2017

# Distributed Optimization Method for Intelligent Control of DC Microgrids

Yuanyuan Fan
*University of South Carolina*

# Distributed Optimization Method For Intelligent Control Of DC Microgrids

by

Yuanyuan Fan

Bachelor of Science
He Fei University of Technology 2004
Master of Science
Shanghai Jiaotong University 2007

---

Submitted in Partial Fulfillment of the Requirements

for the Degree of Doctor of Philosophy in

Electrical Engineering

College of Engineering and Computing

University of South Carolina

2017

Accepted by:

Herbert L. Ginn, Major Professor

Roger A. Dougal, Committee Member

Andrea Benigni, Committee Member

Edward P. Gatzke, Committee Member

Cheryl L. Addy, Vice Provost and Dean of the Graduate School

# Dedication

To the memory of my grandmother, Chunye Li (1927-2015), who is always in my heart.

# Acknowledgments

Firstly, I would like to express my sincere gratitude to my advisor, Dr. Herbert L. Ginn, for his support of my Ph.D study and related research. His valuable advice and guidance helped me in all the time of research. Without his generous support, this dissertation would not have been materialized. Dr. Ginn has supported me not only by providing a research assistantship for several years, but also academically and emotionally through the rough road during this journey.

Besides my advisor, I would like to thank the rest of my advisory committee members, Dr. Roger Dougal, Dr. Andrea Benigni, and Dr. Edward Gatzke, for their time and insightful advice. Their comments help me view my research with various perspectives.

I also want to thank Huaxi Zheng, Qiu Deng, Dan Li, Md Rishad Hossain, and Kang Peng, for their kind support and friendship.

My eternal gratitude to my father, Junming Fan, and to my mother, Youxie He, for their dedication and support during my life. I also would like to thank my husband and my best friend, Yang Gao, for his continuous encouragement through the toughest moments of my life. Finally there are my wonderful children, Evelyn Gao and Ethan Gao, who have given me much happiness and made me better and stronger. I love you all more than anything.

# Abstract

Most of the distributed energy resources, loads and energy storage systems in a DC microgrid are equipped with power electronic converters. With the integration of advanced power electronics devices, a microgrid is able to utilize a broader range of technologies in its design and operation. A key feature of power electronic converter based systems is the ability to direct energy flow within a system with their coordinated operation. A system level control is needed for coordination where converters execute reference points dictated by a system-level control in order to achieve system level goals. System goals can be expressed as a cost function solved by a real-time optimization algorithm. This work develops a framework for the coordinated operation of converters with a distributed optimization method for use in a real-time system-level control system.

In order to validate the optimization based control method developed in this research, a simplified shipboard DC power distribution system is used for case studies. It is an isolated microgrid with converters between all sources of energy and the main buses as well as between all load centers and the main buses. The example cost function used in the study minimizes distribution losses in the DC power system. Initially, the optimization problem is solved using a centralized method in order to provide a baseline for evaluating other schemes. Primal-dual interior point method is applied successfully to provide optimal operating points. The centralized structure relies on one central controller to support the entire system control such that the system is vulnerable to single points of failure and not easily expandable.

To address the robustness and expandability shortcomings, a distributed coor-

dinating optimization algorithm is developed. The coupling constraints formed by nodal current balance result in control variable coupling, therefore, techniques are required to perform an appropriate decomposition. The main task of this dissertation is to develop a practical distributed algorithm via the decomposition of the optimization problem. The method developed here combines dual decomposition and Alternating Direction Method of Multipliers (ADMM) together. This is an iterative based method. By utilizing a decomposition method, the microgrid is partitioned into multiple subsystems. The global target is achieved by interaction of the subsystems which operate on local information. The solutions from the decomposition method and centralized method are compared in diagrams and in numbers using the shipboard DC microgrid test system. Results show that the numerical results from both methods match closely. Analysis of the effect of the number of microgrid subsystem partitions on convergence speed of the decomposition method is also performed.

# Table of Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

## 1.1 Microgrids

In traditional grids, electrical energy is usually generated by regional power plants and then distributed to serve load centers which are often distant from generation sites [3]. Changes are taking place in this pattern because of the increasing emergence of smaller Distributed Energy Resources (DERs). DERs are energy resources that are small scale but located close to loads. They include not only generators and energy storage systems but also, for some classes of systems, advanced power electronics conversion equipment between the energy sources and their surrounding systems [20][30]. The potential of DERs can be more fully developed when organized into a microgrid [33][21][19]. Microgirds are discrete energy systems consisting of DERs and loads capable of operating in parallel with, or independently from, the power grid. Microgrids are designed to support Alternating Current (AC) or Direct Current (DC).

In DC microgrids, sources and loads are generalized into three types: distributed generation units, storage systems, and distributed loads. Firstly, distributed generators are able to be connected to DC systems through converters [23]. This structure provides benefits by combining together different generation types such as photovoltaic cells, micro turbines, and fuel cells. Synchronization between the new energy sources and the DC network is not required, and frequency regulation is not necessary. Secondly, energy storage systems are interconnected through converters to improve

power quality. This plays an important role to balance power between generation and consumption, especially when rapid changes in the load occur to low inertia systems. Finally, the load centers connected in DC mircogrids are also interfaced via DC/DC or DC/AC converters. Stable voltages should be ensured by the control strategies applied on those converters even in the presence of varying loads.

Well-designed DC grids can improve efficiency by decreasing losses associated with conversion stages. For example, Variable Speed Drives (VSDs) powered by DC have 5% less power consumption than those powered by AC due to avoid out of a rectification stage[13]. This is especially important for microgrids such as a ship where a large amount of the generation is used in the propulsion system which is a type of VSD.

Power electronic devices are coordinated by Energy Management Systems (EMSs) to achieve system level goals. The real time data is used for planning and determination of converter operating points. Orders are sent to various converters [22][17] such that an optimum operation of the entire system can be achieved. Some research has been performed on system-level optimal power flow [2][14] most of which focus on the optimization of a traditional AC distribution power system [10][28][29]. The focus of this research is on DC distribution systems with power converters isolating AC power on the sides of generators or loads. There are two basic structures to achieve the system level control: centralized and distributed architectures [17]. In this research, a distributed energy management solution for DC microgrid system is developed.

## 1.2 Statement of Problem and Approach

Developing an optimization method to manipulate the power electronic devices and thereby direct optimal energy flow in the system is desirable to reduce losses. Various optimization techniques can be used to solve nonlinear sets of equations representing the grid. A centralized method using primal dual interior-point algorithm

provides a baseline for comparison. To apply this centralized method, the optimization problem is formulated as a convex programming problem [5] incorporating capacities of generation, batteries, and power electronics devices. Any measured change happening in the system can be immediately handled by the centralized controller, so this method can be efficient. However, it is not very expandable and fault tolerant. Changes in system states require a system-wide recalculation. Here a distributed optimization architecture is proposed.

In this distributed framework, main controllers in subsystems perform local management functions while coordinating with each other to fulfill global objectives. Robustness and scalability are brought into the microgrid by the distributed control structure [9]. Subsystem controllers, as a distributed and autonomous unit towards problem solving, play a key role to inhibit system collapse when one or more of their components fail. The system is more robust since single points of failure are avoided. In addition, the distributed structure provides a base for easy system expansion with intelligent local controllers when load increases. In other words, each controller is allowed to join or leave the system easily.

The distributed control developed in this dissertation incorporates dual decomposition and Alternating Direction Method of Multipliers (ADMM) together. Dual decomposition is based on applying Lagrange multipliers (dual variables) to the coupling constraints. ADMM extends the decomposition idea by using augmented Lagrangian. The optimal solution from decomposition method matches closely to centralized method.

## 1.3 Overview

A shipboard DC power system [7] as an isolated DC microgrid is used in this study to verify the decomposition-based power management. The dissertation is organized as follows.

In chapter 2, a shipboard DC microgrid is electrically analyzed and modeled as an appropriate optimization function and system constraints all identified.

In chapter 3, using a centralized framework, the problem is directly formulated based on the proposed model and solved with the primal-dual interior-point method. This optimization determines the set points of power electronic converters to implement an efficient energy dispatch. The results from this centralized method provides data about attributes of the studied system which can be used as a baseline by decentralized methods.

This research then focuses on distributed optimization in chapter 4. In this study, by partitioning the studied system to three subsystems, mathematical model applicable to decomposition is built up. Then a flow chart detailing the distributed algorithm involving dual decomposition and ADMM is presented.

Simulation results from Matlab Simulink are shown in Chapter 5. More test cases to verify the algorithm convergence are also present in Chapter 5.

# Chapter 2

# System Outline and Electrical Analysis

A shipboard DC power distribution system is used as the object of study. Figure 2.1 shows the layout of this system comprised of two power sources, two load zones, one energy storage system (ESS), and one pulsed load [31][18]. Power converter modules (PCMs) in Figure 2.1 operate as controllable current sources. Their current reference values come from an EMS. The EMS proposed in this dissertation is restricted to the problem of power distribution efficiency. More specifically, in the case of this study, in order to minimize distribution power loss, branch currents are controlled by coordinating the action of PCMs.

An electrical model for this shipboard power system is introduced as shown in Figure 2.2. This steady-state model depicted in this figure is based on the representation of converters by controllable current sources. The figure has also integrated the equivalent transmission line resistances of the shipboard DC system.

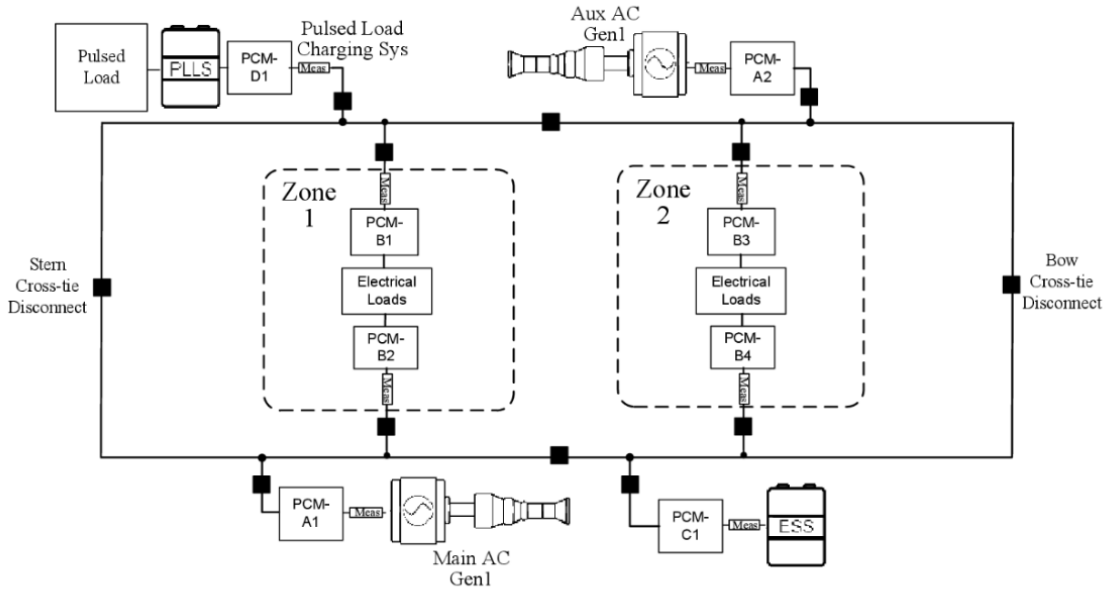


Figure 2.1: Studied shipboard power system with two generators, an ESS, two zones, and a pulsed load charging system

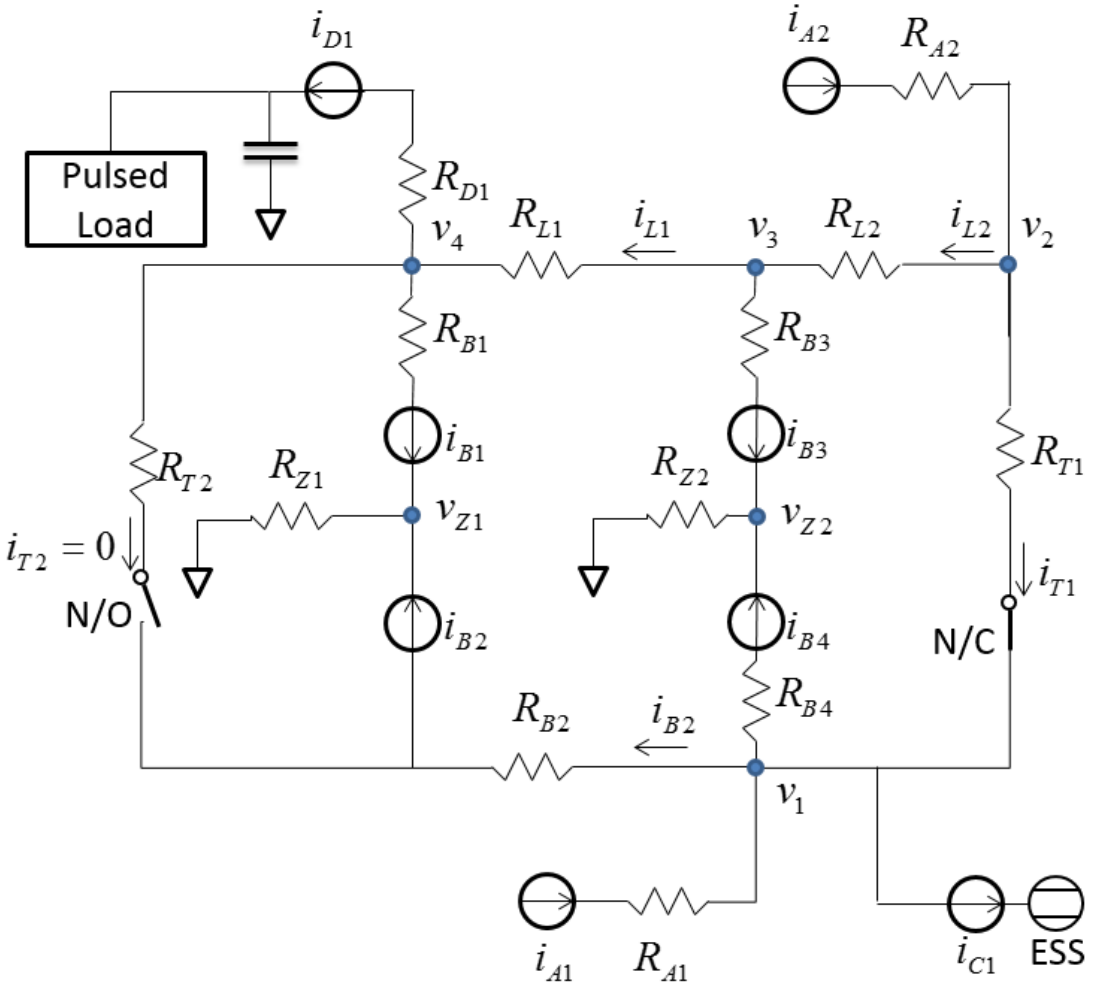


Figure 2.2: System model with current sources

An example objective function is selected as the minimization of power loss, which can be expressed as

$$f = \sum_{m \in D} i_m^2 R_m. \tag{2.1}$$

where $D$ is the set of all branches in the grid. The system shown in Figure 2.2 has six buses. Current equality at each bus gives constraint equations as follows.

$$i_{A1} = i_{B2} + i_{B4} - i_{T1} + i_{C1} \tag{2.2}$$

$$i_{A2} = i_{L2} + i_{T1} \tag{2.3}$$

$$i_{L1} = i_{D1} + i_{B1} \tag{2.4}$$

$$i_{L2} = i_{L1} + i_{B3} \tag{2.5}$$

$$i_{B1} + i_{B2} = i_{Load1} \tag{2.6}$$

$$i_{B3} + i_{B4} = i_{Load2} \tag{2.7}$$

Ohms' Law defines the relationship among variables on the linear branches. This brings about three more constraints.

$$v_2 - v_1 = R_{T1} i_{T1} \tag{2.8}$$

$$v_2 - v_3 = R_{L2} i_{L2} \tag{2.9}$$

$$v_3 - v_4 = R_{L1} i_{L1} \tag{2.10}$$

The power injected in system by the two generators can be calculated in terms of currents and voltages as:

$$P_{A1} = i_{A1} v_1 + i_{A1^2} R_{A1} \tag{2.11}$$

$$P_{A2} = i_{A2} v_2 + i_{A1^2} R_{A2}. \tag{2.12}$$

Voltage stabilization function has been embedded into the lower level control system in which voltage $v_1$ is controlled to be 500V, and voltages $v_{Z1}$ and $v_{Z2}$ are maintained at 400V. As illustrated in the equation (2.13), based on the flow conservation, total input current is equal to the sum of load currents in per unit system. This can also be achieved by adding equations (2.2) to (2.7) together. The load in this system includes zonal loads, a pulsed power load, and an energy storage system which can operate as a load or source at different times depending on the system requirements. Furthermore, due to the existence of charging capacitor on pulsed power load, transients are handled such that a sudden current change is not directly imposed on the system but only a steady charging current $i_{D1}$.

$$i_{A1} + i_{A2} = i_{C1} + i_{D1} + i_{Load1} + i_{Load2} \tag{2.13}$$

Table 2.1: Inequality Constraints of Test Systems

| Capacities On Test Systems [11] | Inequality constraints |
|---|---|
| 30kW main generator<br>per 10kW auxiliary generator | $i_{A1} \leq 30/15$<br>$i_{A2} \leq 10/15$ |
| 10kW ship service loads per zone | $i_{B1}$,$i_{B2}$,$i_{B3}$,$i_{B4} \leq 10/15$ |
| 350kJ Pulsed Load local storage<br>with a peak power draw of 20kW | $i_{D1} \leq 20/15$ |
| Energy Storage System: 10kW converter<br>attached to a 1.2MJ storage system | $\|i_{C1}\| \leq 10/15$ |

In practical settings, generator capabilities have limits and transmission line currents are also bounded to avoid overloading of the PCMs. Therefore values of those variables should be defined within specific limits. Moreover the laboratory-scale test system is scaled down by a factor of approximation of 1/500 as compared to a notional ship system [11]. Table 2.1 summarizes specific current limits on test system and its corresponding constraints on control variables in per unit system with base voltage 500V and base power 15kW. Notice every quantity used in the study is normalized. So when rated voltage of an equipment is equal to base voltage, $i^{p.u.}$ has the same value of $p^{p.u.}$ since

$$i^{p.u.} = \frac{p^{p.u.}}{v^{p.u.}} = \frac{p^{p.u.}}{1} = p^{p.u.} \tag{2.14}$$

# Chapter 3

# Optimization Formulation with Centralized Method

Energy management for shipboard integrated power system is essentially a global optimization problem whose objective is to determine the flow on each line in such a way that transmission loss is minimized [17]. The solution can be viewed as a sequence of commands to the control systems of PCMs at each instant that leads to the optimization goal. Here, centralized control is implemented mainly because of the higher efficiency as compared to distributed control methods which rely on iteration among controllers. Centralized control methods require the knowledge of the entire topology of the system and the system limits ahead since system equations are created based on this information [18]. In fact, the solutions obtained by solving the equations are greatly affected by how to formulate the problem. If the overall system model is developed accurately enough and equations can be solved in real-time, good control performance can be expected using centralized control methods.

## 3.1 Optimization Formulation

In Chapter 2 system, variables are comprised of three sets of components: currents on each branch, voltage on each bus, and generator output power values $P_{A1}$ and $P_{A2}$. However, from equations (2.8) - (2.12), it can be concluded that all system voltages and generation values totally depend on currents around the circuit. It is feasible to ignore them to reduce the variable numbers and only take into account

currents as unknown problem variables. Therefore, the problem variables shrink to $x = (i_{A1}, i_{A2}, i_{B1}, i_{B2}, i_{B3}, i_{B4}, i_{L1}, i_{L2}, i_{C1}, i_{T1})$. In this way, the loss function shown as equation (2.1), the equality constraints defined by equations (2.2) - (2.7), and inequality constraints following Table 2.1 are in the form of

$$\begin{aligned} &\textit{minimize} \quad f(x) = x^T R x \\ &\textit{subject to} \quad Ax = b \\ &\qquad\qquad x_{min} \leq x \leq x_{max} \end{aligned} \tag{3.1}$$

where $R = diag(R_{A1}, R_{A2}, R_{B1}, R_{B2}, R_{B3}, R_{B4}, R_{L1}, R_{L2}, 0, R_{T1})$,

$$A = \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & -1 & 0 & 1 & -1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$b = \begin{bmatrix} 0 & 0 & 0 & 0 & i_{Load1} & i_{Load2} \end{bmatrix}^T,$$

$x_{min}$ and $x_{max}$ are the bounds defined by Table 2.1. This formulated problem is convex because it consists of the minimization of a quadratic form function subject to linear constraints [5]. After the optimization equations are solved in the management system, a subset of the solutions is transmitted to low level control of PCMs as current reference values. The control hierarchy is shown in [11]. In the case of this research, the values that are dispatched to local controls include the value of bus-tie current as well as the sharing ratios for each load zonal current. Here, load sharing ratios describe how loads in each zone are shared between the two feeding branches. One may assume that one of the bus-tie breakers is open, as in the normal operational of the system. Therefore, the target is formed to issue three unknown values including two zonal load ratios and one bus-tie current, which lead to minimum transmission

loss.

## 3.2 Primal-Dual Interior-point Optimization Method

Authors in [16] give rise to a great breakthrough in the field of interior-point methods [35][36][34][38] which is an important part in conventional optimization method. This field has experienced rapid development. A primal-dual algorithm performs better on practical problems than other interior-point methods [37]. Among the implementation of primal-dual interior-point algorithm, the free code MATPOWER [40] is used here. In this tool, loss function and its gradient, constraints and their gradients, the Hessian of the Lagrangian, and an unnecessarily feasible starting point are required as solver inputs. The problem formulated above can be further generalized to:

$$\begin{aligned} &minimize \quad f(x) = x^T R x \\ &subject\ to \quad G(x) = 0 \\ &\qquad\qquad H(x) \leq 0 \end{aligned} \tag{3.2}$$

where $f(x)$ is objective function, $G(x)$ and $H(x)$ are respectively termed as equality and inequality constraints that are defined as:

$$\begin{aligned} G(x) &= Ax - b \\ H(x) &= \begin{bmatrix} -x + x_{min} \\ x - x_{max} \end{bmatrix} \end{aligned} \tag{3.3}$$

By adding slack variable $Z$ and using barrier function with the parameter of perturbation $\gamma$, equation 3.2 is formed as:

$$
\begin{aligned}
&\textit{minimize} \quad \left[f(x) - \gamma \sum_{m=1}^{n_i} ln(Z_m)\right] \\
&\textit{subject to} \quad G(x) = 0 \\
&\qquad\qquad\quad H(x) + Z = 0 \\
&\qquad\qquad\quad Z > 0
\end{aligned}
\tag{3.4}
$$

where $n_i$ is the number of inequality constraints. For a given number $\gamma$ the Lagrangian with dual term $\lambda$ and $\mu$ is

$$
L^{\gamma}(x, Z, \lambda, \mu) = f(x) + \lambda^T G(x) + \mu^T (H(x) + Z) - \gamma \sum_{m=1}^{n_i} ln(Z_m) \tag{3.5}
$$

The Karush-Kuhn-Tucker(KKT) condition, first-order optimality condition, is satisfied when:

$$
\begin{aligned}
&F(x, Z, \lambda, \mu) = 0 \\
&\qquad\qquad Z > 0 \\
&\qquad\qquad \mu > 0
\end{aligned}
\tag{3.6}
$$

where $F(x, Z, \lambda, \mu) = \begin{bmatrix} f_x^T + G_x^T \lambda + H_x^T \mu \\ \mu Z - \gamma e \\ G(x) \\ H(x) + Z \end{bmatrix}$.

In primal-dual interior-point method, Newton's method is always applied to solve KKT conditions. In a matrix form:

$$
\begin{bmatrix} L_{xx}^{\gamma} & 0 & G_x^T & H_x^T \\ 0 & \mu & 0 & Z \\ G_x & 0 & 0 & 0 \\ H_x & I & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta Z \\ \Delta \lambda \\ \Delta \mu \end{bmatrix} = - \begin{bmatrix} L_x^{\gamma T} \\ \mu Z - \gamma e \\ G(x) \\ H(x) + Z \end{bmatrix} \tag{3.7}
$$

where $L_{xx}^{\gamma}$ is the Hessian, $e = [1, ..., 1]^T$. After obtaining $\Delta x$, $\Delta Z$, $\Delta\gamma$, $\Delta\mu$, by using (3.8), $\alpha_p$,$\alpha_d$ are computed as follows in order to maintain strict feasibility.

$$\begin{aligned} \alpha_p &= \min\left(\xi \min_{\Delta Z_m<0}\left(-\frac{Z_m}{\Delta Z_m}\right), 1\right) \\ \alpha_d &= \min\left(\xi \min_{\Delta \mu_m<0}\left(-\frac{\mu_m}{\Delta \mu_m}\right), 1\right) \end{aligned} \tag{3.8}$$

where scalar $\xi$ is set to 0.99995 in code. Following (3.7) and (3.8) variables can be updated as (3.9).

$$\begin{aligned} x &\leftarrow x + \alpha_p \Delta x \\ Z &\leftarrow Z + \alpha_p \Delta Z \\ \lambda &\leftarrow \lambda + \alpha_d \Delta \lambda \\ \mu &\leftarrow \mu + \alpha_d \Delta \mu \end{aligned} \tag{3.9}$$

After each iteration, $\gamma$ is updated to $\sigma\frac{Z^T\mu}{n_i}$ (where $\sigma$ is set to 0.1) due to this perturbation parameter needs to converge to zero.

## 3.3 Simulation and Results

During the simulation, after loads (zonal loads, ESS, and Pulsed power load) are switched on into the circuit, quantities are measured at each instant to update the equations. Then, the algorithm determines the optimal current distribution that minimizes the loss, while providing the loads and respecting the constraints. Subsequently, lower level control tries to follow the equivalence factors calculated from optimal distribution until next update arrives. Although one cycle of optimization calculation takes a few milliseconds, which is a reasonable time frame, a period of 45ms is given to the algorithm to deal with problem, because in real world communication delay should also be considered.

The following scenario is used in the simulation to assess the approach. Initially, both zones are loaded at 10kW and a power of 10kW is moved across the bus tie towards the aux generator which is running at 40% rating. The optimization command

is given at 1s, and the values of currents are set accordingly. The pulsed power load is activated at 2.5s. This in turn activates its charging system. The energy storage system is also connected at 2.5s. After almost half a second, the energy storage system is shut down by its control system, since the State Of Charge (SOC) hits its lower limit. At 9s, the Zone1 load drops off by 50% and the Zone2 load decreases to 25%. Three seconds later, at 12s, an increment is made at Zone2 to 75% of its initial value, 7.5kW, while the Zone1 load remains unchanged.

The simulation results of the periodically refreshed current values over a 15 second interval are plotted in Figure 3.1. These orders from systematic optimization are implemented in real-time. It is worth noticing that the Zone1 sharing ratio has no significant change because the current splitting ratio essentially is not sensitive to load variation. But due to limited generator capacities, an obvious ratio change is imposed to Zone2. Additionally, quantities of system resistances especially bus tie breaker resistance highly affects the optimization results.

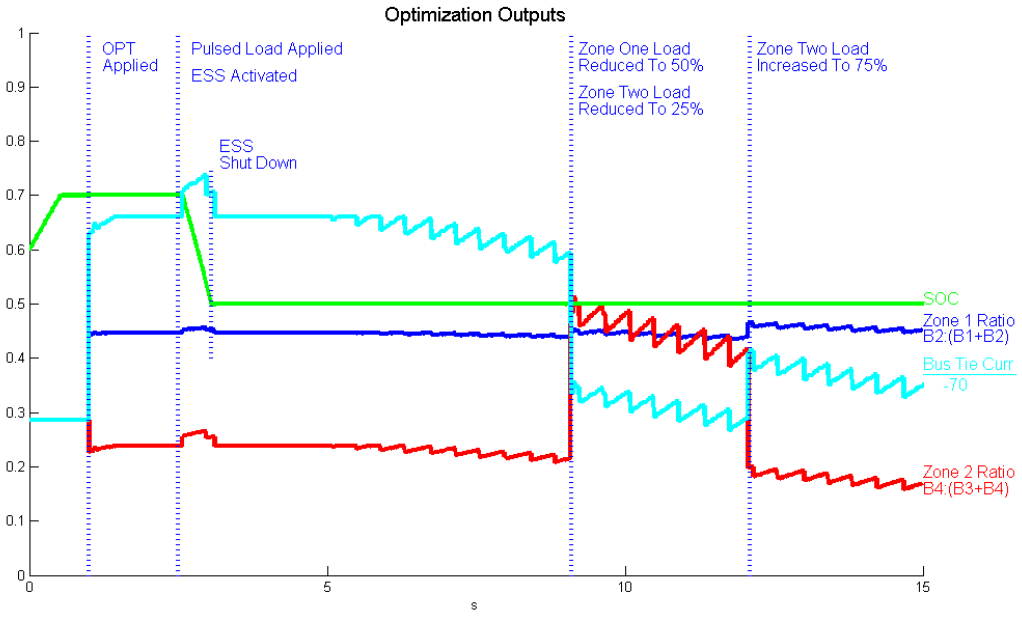


Figure 3.1: Commands from centralized system-level control

Commands are in the form of load sharing ratio, bus tie current or storage current

Figure 3.2 compares the system losses between the system with optimization applied and the system with fixed sharing rules. The system loss is decreased when optimization is applied.

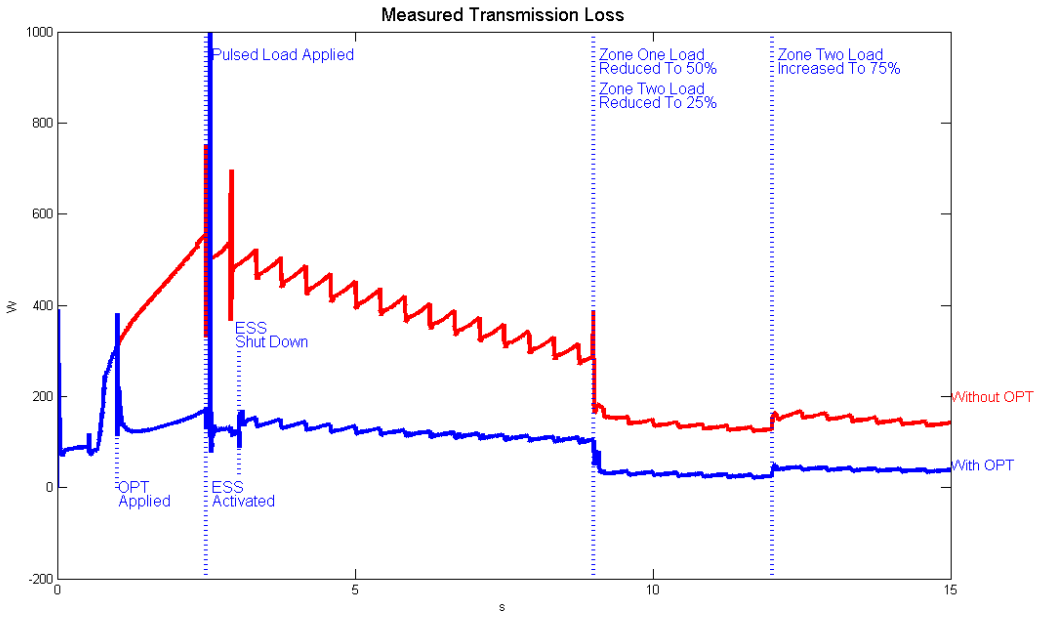


Figure 3.2: Losses with and without centralized optimization

Solutions of optimization which are sets of current flow in branches are shown in Figures 3.3 and 3.4. It is seen that actual currents of PCM-A1 and PCM-A2 are effectively controlled to stay within generation current limits except for a few spikes in the current. Furthermore, bus voltages depicted in Figure 3.5 are all maintained within acceptable tolerance.

Figure 3.6 shows the values of current flows. In this figure, [CURRENT] / [CURRENT] / [CURRENT] / [CURRENT] shows current flows at state 1/2/3/4. State 1 is when pulsed load is activated and zones are at full load (10kW for each zone). State 2 happens when ESS is shut down. State 3 is when Zone1 load is reduced to 50% and zone two being reduced to 25%. At state 4 Zone2 load is increased to 75%. This figure depicts that optimum dispatch tends to supply a load from the generation unit which is electrically closer to it until the limits of that unit collegese. To ensure

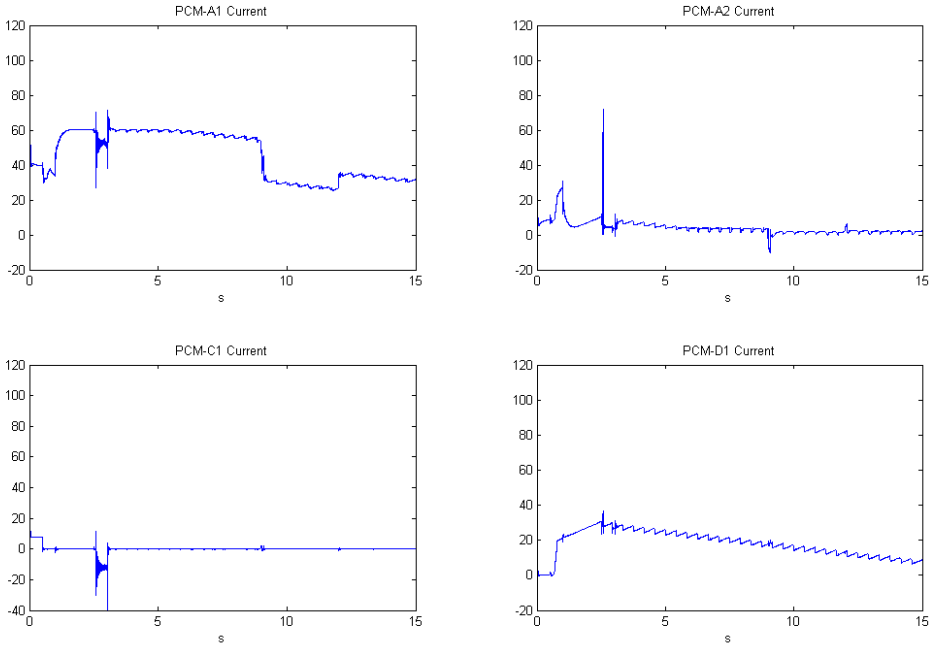


Figure 3.3: Current flows on bus connected converters under centralized system-level control

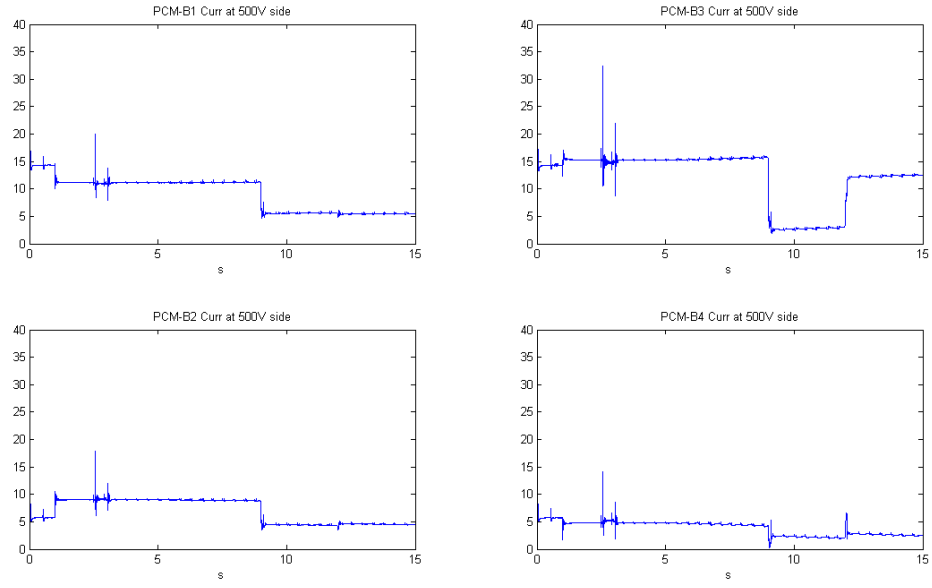


Figure 3.4: Current flows on zonal converters under centralized system-level control

that flows are within the capabilities of each generator, bus tie current is adjusted as needed.

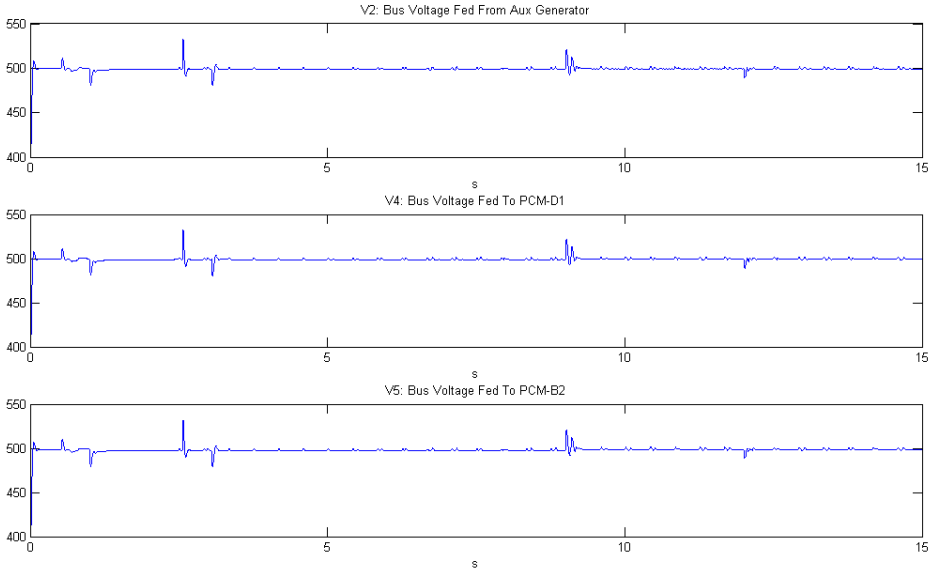


Figure 3.5: Measured bus voltages

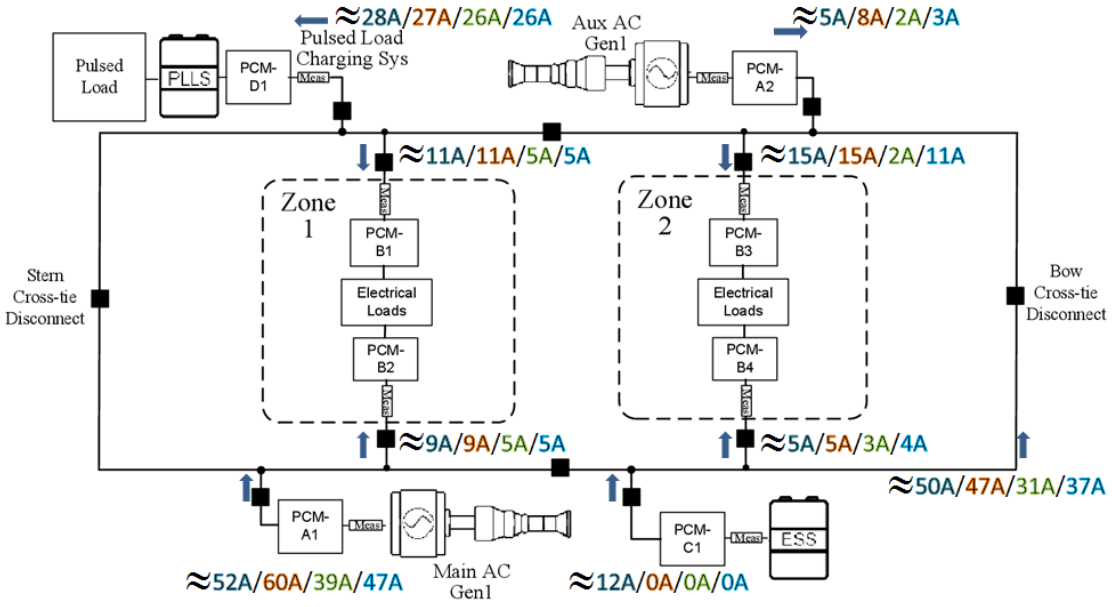


Figure 3.6: Approximate numerical results

To demonstrate the validity of primal-dual interior-point algorithm in minimization of loss, one randomly selected instance time is considered at 2.9s. At this moment, pulsed load is applied and ESS is actively supplying power which introduce $i_{D1}$ to be around 30A, and $i_{C1}$ be -12A. Because there are three free variables in the

formulated problem, one randomly chosen variable is set to be the value originated from the optimal operation ($i_{L1}$ is 40.3636A). The other two variables are able to form a three-dimensional Cartesian coordinate with loss as shown in Figure 3.7. It can be seen that the shape of loss is paraboloid as two currents vary over the surface. The minimum power loss point calculated above from the algorithm happens at $i_{A1} = 53.9576$A and $i_{L2} = 55.0497$A which depicts a very close agreement with the Figure 3.7. This can be repeated by changing the chosen instance of time or chosen variables for the two dimensions, and the results from 3-d figures are consistant with the optimum values from the optimization algorithm.

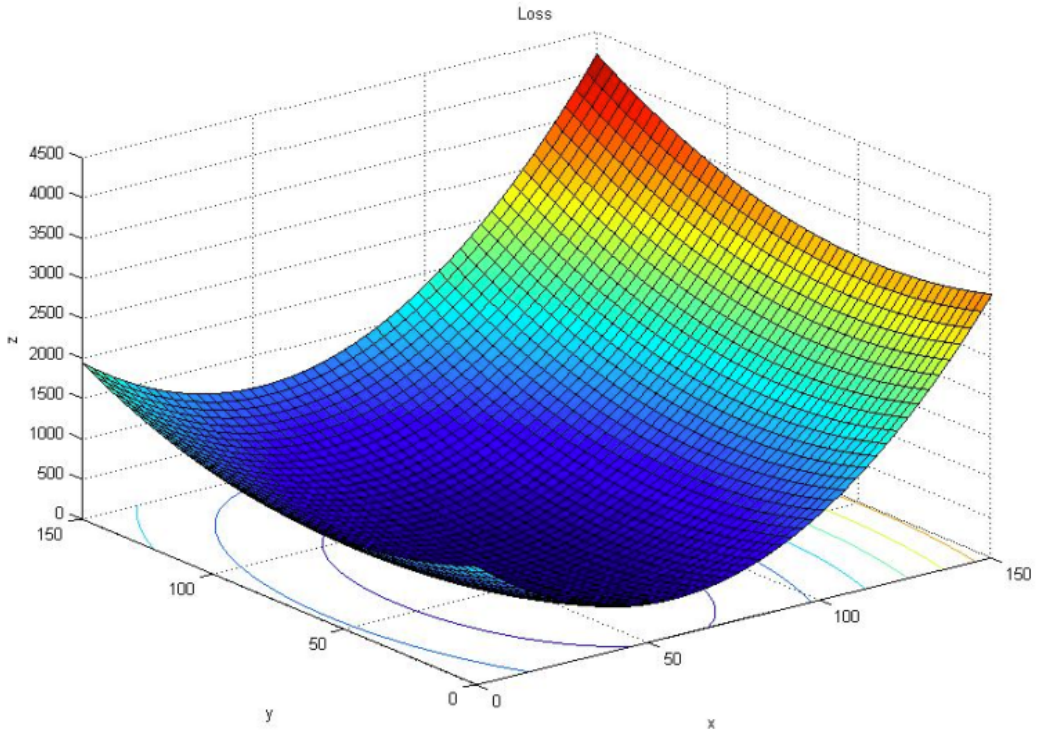


Figure 3.7: Power loss with variations of $i_{A1}$ and $i_{L2}$ ($x = i_{A1}$, $y = i_{L2}$, $z = loss$)

# Chapter 4

# Optimization Formulation with Distributed Method

Because the centralized structure relies on one central controller to support the entire system, it is vulnerable to single points of failure. To solve this problem, a decentralized system is developed. In a decentralized system, each component is equally responsible for contributing to the global optimum, so it is more robust and naturally eliminates single points of failure. In such a system, instead of gathering the information from the whole system, controllers only use local information. This type of controlling scheme is called distributed control. A distributed control algorithm is developed here to perform converter coordination in order to optimize the cost function presented in equation (2.1)

From the point of view of calculation time, if the electrical system has greater size, it is not practical to have the planning and optimization done by a single controller. This is especially true in the case computational time grows more than linearly with the problem size. A distributed method is proposed to handle this problem. Here, a decomposition method is used to iterate among subsystems to achieve a solution. Unlike centralized coordination in which local controllers receive set points from single specific central controller, here controllers make decisions locally. But still, the major task of each controller is not necessarily to maximize the budget of its corresponding unit but to improve overall performance of the system.

This work is based on the research of decomposition optimization [39][1][8][25].

The main idea of distributed method is to reduce the whole problem into individual small problems, introducing small cost functions that depend only on local variables [26][27]. The relation among the small problems are the public variables. They are synced in between each optimization step via a reliable communication system.

As the case study, the shipboard power system is used to demonstrate how this approach works in real time with the results from centralized method as the ground truth.

## 4.1 Decomposition Framework of General Problem

Consider a distributed optimization problem [6] with $K$ subsystems, each of the subsystems contains one or more power converter modules with local information collection and distribution. In subsystem $i$, $x_i \in R^{n_i}$ and $y_i \in R^{P_i}$ are used to denote the private and public variables, $f_i : R^{n_i} \times R^{p_i}$ is a local objective function. These subsystems are coupled through public variables, which have equal individual scalar components. Thus the overall system has the objective function

$$f(x, y) = f_1(x_1, y_1) + f_2(x_2, y_2) + ... + f_K(x_K, y_K). \tag{4.1}$$

With the local variables constrained by the set $C_i \subseteq R^{n_i} \times R^{p_i}$, it follows that $(x_i, y_i) \in C_i$. Furthermore, let $N$ be the number of nets in the system. A net is a hyperedge which connects two or more subsystems in the hypergraph. A net corresponds to a public variable, so $z \in R^N$ is introduced to present common values of the public variables on the nets. The set of all public variables is denoted by $y \in R^p$, where $y = (y_1, ..., y_K)$ and $p = \sum_{i=1}^{K} p_i$. Then the relationship between $y$ and $z$ can be described as

$$y_i = E_i z \tag{4.2}$$

$$\text{where} \quad E_{ij} = \begin{cases} 1 & (y)_i \text{ is in net,} \\ 0 & \text{Otherwise.} \end{cases}$$

Here, $(y)_i$ denotes the $i$th component of vector $y$. The $i$th row block of $E$ is denoted as $E_i \in R^{p_i \times N}$ which specifies the complete net list. The problem given by (3.2) now has the form:

$$\begin{aligned} min. \quad & f(x,y) = f_1(x_1,y_1) + f_2(x_2,y_2) + ... + f_k(x_k,y_k) \\ s.t. \quad & (x_i,y_i) \in C_i \quad i = 1,...,K, \\ & y_i = E_i z \qquad i = 1,...,K. \end{aligned} \tag{4.3}$$

The problem now is in the form of a distributed architecture.

## 4.2 Dual Decomposition Algorithm

Dual decomposition, which provides a mechanism to deal with problems in parallelized manner [6][32][15], is used here to solve the DC microgrid optimization problem in a distributed form as in (4.3). By incorporating constraints into objectives, the Lagrangian of the problem can be formed as

$$\begin{aligned} L(x,y,z,v) &= \sum_{i=1}^{K} f(x_i,y_i) + v^T(y - Ez)_i \\ &= \sum_{i=1}^{K} (f_i(x_i,y_i) + v_i^T y_i) - v^T E z, \end{aligned} \tag{4.4}$$

where $v = (v_1,...,v_k)^T$ is the Lagrange multiplier associated with $y = Ez$. The Lagrange dual function $g(v)$ is the optimal cost under the prices $v$. To ensure $g(v) > -\infty$, $E^T v = 0$ is drawn as a problem constraint. After $v^T E z$ is diminished, the items left in the Lagrangian is completely separable.

Define $g_i(v_i)$ as the solution of problem

$$\begin{aligned} min. \quad & f_i(x_i,y_i) + v_i^T y_i \\ s.t. \quad & (x_i,y_i) \in C_i. \end{aligned} \tag{4.5}$$

The dual problem can then be expressed as

$$\begin{aligned} max. \quad & g(v) = \sum_{i=1}^{K} g_i(v_i) \\ s.t. \quad & E^T v = 0. \end{aligned} \tag{4.6}$$

With this master problem solved by projected subgradient method, the dual decomposition algorithm [6] is shown below.

**Given** initial price vector $v$ that satisfies $E^T v = 0 (e.g., v = 0)$

**Repeat**

*Optimize subsystems (separately).*

Solve subproblem (4.5) to obtain $x_i$,$y_i$.

*Compute average of public variables over each net.*

$\hat{z} := (E^T E)^{-1} E^T y$

*Update the dual variables.*

$v := v + \alpha_k (y - E\hat{z})$.

**Until** stopping criterion satisfied

## 4.3 Alternating Direction Method of Multipliers

Alternating Direction Method of Multipliers (ADMM) is a simple but practical decentralized algorithm. It is based on augmented lagrangians and advanced with dual decomposition. The theoretical part of the algorithm has been developed in the eighties. Nowadays, it has started to support distributed calculation of big data and has broad application in statistics and machine learning. At first, a optimization problem with the following form is considered.

$$\begin{aligned} min. \quad & f(x) + g(y) \\ s.t. \quad & Ax + By = c, \end{aligned} \tag{4.7}$$

where $x$ and $y$ are control variables. Here, constraints are affine. By incorporating constraints into objective function, the augmented lagrangian of this problem is

$$L_\alpha(x, y, v) = f(x) + g(y) + v^T(Ax + By - c) + \alpha/2\|Ax + By - c\|_2^2, \tag{4.8}$$

where $\alpha$ is the penalty vector. The introduction of augmented lagrangian yields convergence without objective function being strict convex or finite. ADMM iteration can be updated as follows.

$$x^{k+1} := \underset{x}{\operatorname{argmin}}\, L_\alpha(x, y^k, v^k), \tag{4.9}$$

$$y^{k+1} := \underset{y}{\operatorname{argmin}}\, L_\alpha(x^{k+1}, y, v^k), \tag{4.10}$$

$$v^{k+1} := v^k + \alpha(Ax^{k+1} + By^{k+1} - c). \tag{4.11}$$

Here, $x$ and $y$ are alternating descended which is where the name comes from. Then lagrangian multipliers are updated. If $x$ and $y$ can be efficiently optimized locally, ADMM is particularly practical [4].

## 4.4 Consensus Optimization

Let's consider the objective function be decomposed into $N$ parts as:

$$min. \quad f(x) = \sum_{i=1}^{N} f_i(x), \tag{4.12}$$

where $f_i(x)$ are all local functions. This problem can be reformulated to fit ADMM form. It is also known as global variable consensus optimization as follows.

$$\begin{aligned} min. \quad & f(x) = \sum_{i=1}^{N} f_i(x_i) \\ s.t. \quad & x_i - y = 0, i = 1, ..., N. \end{aligned} \tag{4.13}$$

The updates from applying ADMM on this problem is the following:

$$x_i^{k+1} := \underset{x_i}{\operatorname{argmin}}(f_i(x_i) + v_i^{kT}(x_i - y^k) + (\alpha/2)\|x_i - y^k\|_2^2), \tag{4.14}$$

$$y^{k+1} := \frac{1}{N}\sum_{i=1}^{N}(x_i^{k+1} + (1/\alpha)v_i^k), \tag{4.15}$$

$$v_i^{k+1} := v_i^k + \alpha(x_i^{k+1} - y^{k+1}). \tag{4.16}$$

The above updates are able to be simplified to the following when using $\sum_{i=1}^{N} v_i^k = 0$.

$$x_i^{k+1} := \underset{x_i}{\operatorname{argmin}}(f_i(x_i) + v_i^{kT}(x_i - \bar{x}^k) + (\alpha/2)\|x_i - \bar{x}^k\|_2^2), \tag{4.17}$$

$$v_i^{k+1} := v_i^k + \alpha(x_i^{k+1} - \bar{x}^{k+1}), \tag{4.18}$$

where $\bar{x}^k = (1/N)\sum_{i=1}^{N} x_i^k$.

### 4.5 Problem Formulation

The shipboard power system with two zonal loads is used to demonstrate how the approach works in real time.

## Splitting Graph Structure

The first step of distributed control is to decompose the problem into subproblems (smaller ones) by exploring the network structure [24]. The corresponding graph of the studied power distribution system has vertices and edges induced in Figure 4.1. This graph merely focuses on the distribution network of the original system. The vertices represent the buses who distribute, consume, or generate power. $V_{v1}$, $V_{v2}$, $V_{v3}$, and $V_{v4}$ as power distributing nodes, $V_{Source1}$, $V_{Source2}$, $V_{ESS}$, $V_{PulseLoad}$, $V_{Load1}$, and $V_{Load2}$ as power generating or consuming nodes are all elected as vertices. Vertices are connected by edges. Here, the edges represent power cables with the assumption that resistances of converters are ignored. The branches that have PCMs equipped are treated as independent buses. Other dependent currents are enforced by controlling the independent currents. In this system, the number of independent circuit equations is six and the number of unknown variables is nine, so three free branch variables need to get reference values from system-level control. In this work, the current through PCM-B1, PCM-B3, and PCM-A2 are chosen as independent variables. Thus

these three PCMs behave like local main controllers. Figure 4.1 is the split graph according to this arrangement. Please note that it is absolutely workable to choose other controller combinations like PCM-B2, PCM-B3, and PCM-A1. This feature also provides a powerful support to system fault tolerance. For example PCM-B2 is totally capable of being a backup of PCM-B1 when emergency happens.

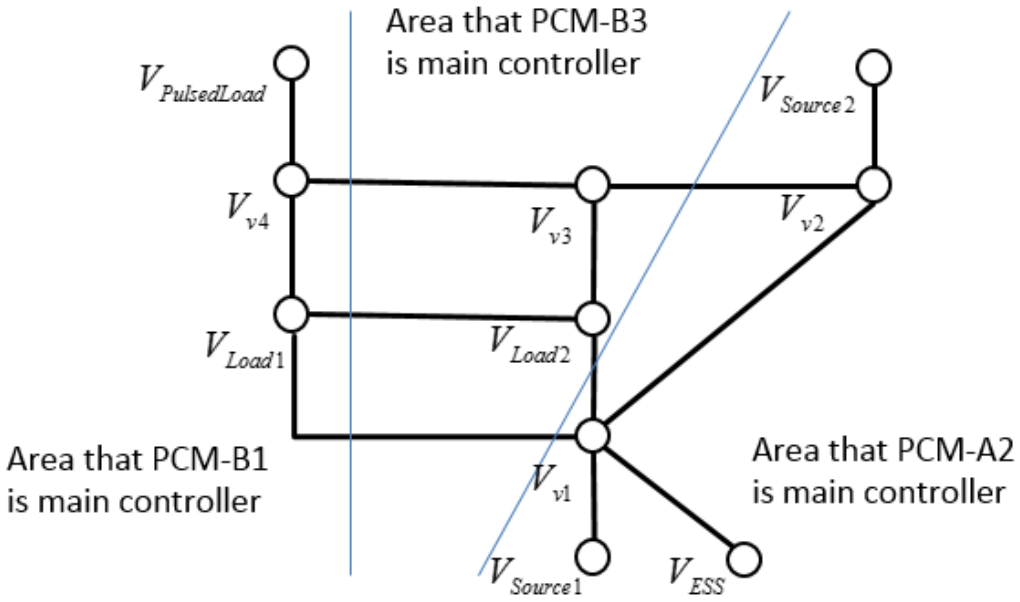


Figure 4.1: Split graph of system

A pre-planned strategy for power grid splitting is necessary. A good partition of the graph can significantly reduce the amount of communication between different areas [12]. Here this system is partitioned into three subsystems as shown in Figure 4.1 . One is for two generators and other two are for two load-zones. This partition method is suitable for this shipboard system because the generator two is just the backup of the main generator. This microgrid works as a self-sufficient system and most loads are fed from the main generator. Due to the low capacity of the backup power source, two generators are put into the same control subsystem. Besides, partition is based on each zonal load area because all zonal loads are running in parallel and they are independent to each other. If each zonal load is governed by its

own controllers, then the extension of the load zones would have no impact on the existing programs.

## Decomposed Problem

With currents as variables and current flows as constraints, the centralized problem is formulated in (2.1) - (2.7). These functions are separated according to the partition pattern mentioned above. Each elected main controller deals with its corresponding subproblem. The overall centralized problem is equivalent to the addition of three small ones shown below.

**PCM-B1**:

$$\begin{aligned} min. \quad & f_{B1} = i_{B1}^2 R_{B1} + i_{B2}^2 R_{B2} + i_{L1}^2 R_{L1} + i_{D1}^2 R_{D1} \\ s.t. \quad & i_{B1} + i_{B2} = i_{Load1}, \\ & i_{B1} - i_{L1} = -i_{D1} \end{aligned} \tag{4.19}$$

where $i_{B1}$, $i_{B2}$, and $i_{L1}$ are variable. The constraint functions can be written in the form of $A_1 x = b_1$, here

$$A_1 = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & -1 \end{bmatrix}, \; b_1 = \begin{bmatrix} i_{Load1} \\ -i_{D1} \end{bmatrix}.$$

**PCM-B3**:

$$\begin{aligned} min. \quad & f_{B3} = i_{B3}^2 R_{B3} + i_{B4}^2 R_{B4} + i_{L2}^2 R_{L2} \\ s.t. \quad & i_{B3} + i_{B4} = i_{Load2}, \\ & i_{B3} + i_{L1} - i_{L2} = 0 \end{aligned} \tag{4.20}$$

where $i_{B3}$, $i_{B4}$, $i_{L1}$, and $i_{L2}$ are variable. As in the form of $A_2 x = b_2$, we have

$$A_2 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & -1 \end{bmatrix}, \; b_2 = \begin{bmatrix} i_{Load2} \\ 0 \end{bmatrix},$$

**PCM-A2**:

$$
\begin{aligned}
&min. \quad f_{A2} = i_{A1}^2 R_{A1} + i_{A2}^2 R_{A2} + i_{C1}^2 R_{C1} + i_{T1}^2 R_{T1} \\
&s.t. \quad i_{A1} - i_{B2} - i_{B4} - i_{C1} + i_{T1} = 0, \\
&\qquad i_{A2} - i_{L2} - i_{T1} = 0
\end{aligned} \tag{4.21}
$$

where $i_{A1}$, $i_{A2}$, $i_{B2}$, $i_{B4}$, $i_{C1}$, $i_{L2}$, and $i_{T1}$ are variable. As in the form of $A_3 x = b_3$, we have

$$
A_3 = \begin{bmatrix} 1 & 0 & -1 & -1 & -1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & -1 & -1 \end{bmatrix}, \; b_3 = \begin{bmatrix} 0 \\ 0 \end{bmatrix},
$$

In such a way, the whole problem is partitioned into three subproblems.

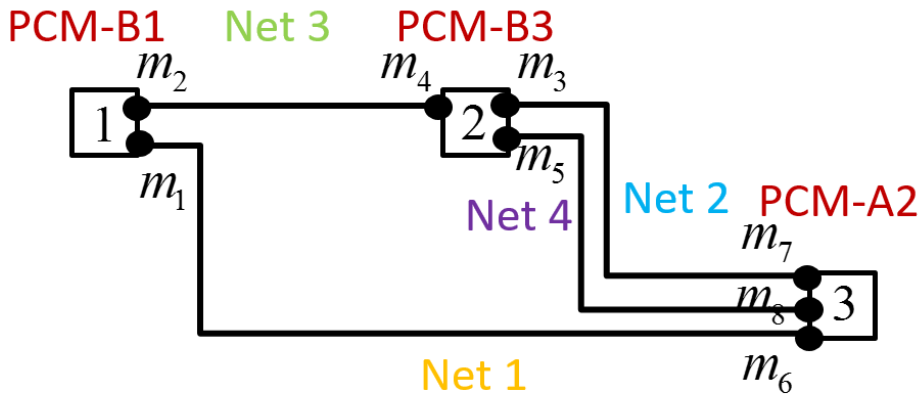


Figure 4.2: System hypergraph

In Figure 4.2, a simple hypergraph is used to represent this decomposition structure. The problems with PCM-B1, PCM-B3, and PCM-A2 are indexed by subproblem 1, 2, and 3 respectively as shown. The nodes, which are shown as squares, are associated with local subproblems consisting of objective and constraints. The nets that connect two or more boxes are associated with complicating variables. Evidently there is some coupling between variables. The coupling variables between subproblem 1 and 2 is $i_{L1}$, the coupling variables between subproblem 2 and 3 are $i_{B4}$ and $i_{L2}$, the coupling variables between subproblem 1 and 3 is $i_{B2}$. In total, the complicating variables in the networks are $i_{B2}$, $i_{B4}$, $i_{L1}$, and $i_{L2}$. Based on the fact that variables

of the master problem are the complicating variables of the original problem, this master problem is formed as follows.

$$
\begin{aligned}
min. \quad & f_{B1}(i_{B1}, i_{B2}, i_{L1}) + f_{B3}(i_{B3}, i_{B4}, i_{L1}, i_{L2}) \\
& \quad + f_{A2}(i_{A1}, i_{A2}, i_{B2}, i_{B4}, i_{C1}, i_{L2}, i_{T1}) \\
s.t. \quad & (i_{B1}, i_{B2}, i_{L1}) \in C_1 \\
& (i_{B3}, i_{B4}, i_{L1}, i_{L2}) \in C_2 \\
& (i_{A1}, i_{A2}, i_{B2}, i_{B4}, i_{C1}, i_{L2}, i_{T1}) \in C_3
\end{aligned} \tag{4.22}
$$

The problem can be further written as

$$
\begin{aligned}
min. \quad & f_{B1}(i_{B1}, m_1, m_2) + f_{B3}(i_{B3}, m_3, m_4, m_5) \\
& \quad + f_{A2}(i_{A1}, i_{A2}, m_6, m_7, m_8, i_{T1}) \\
s.t. \quad & (i_{B1}, m_2, m_1) \in C_1 \\
& (i_{B3}, m_4, m_3, m_5) \in C_2 \\
& (i_{A1}, i_{A2}, m_8, m_6, m_7, i_{T1}) \in C_3 \\
& m_1 = m_6 \\
& m_2 = m_4 \\
& m_3 = m_7 \\
& m_5 = m_8
\end{aligned} \tag{4.23}
$$

In order to reach the desired form, public variables are collected for each subsystem. In our case, subsystem 1 has public variable $y_1 \in R^2$ and local variable $x_1 \in R$. Subsystem 2 has public variable $y_2 \in R^3$ and local variable $x_2 \in R$. Subsystem 3 has public variable $y_3 \in R^3$ and local variable $x_3 \in R^4$. There are four nets here associated with coupling constrains, the consistency constraint $m_1 = m_6$ is expressed for net 1, $m_3 = m_7$ is represented for net 2, $m_2 = m_4$ is for net 3 and $m_5 = m_8$ is for net 4. In other words, the public variables of subsystem 1 and 3 should be equal on the account that they both represent $i_{L1}$ physically. For the same reason, other three coupling constrains can be achieved. Further more, all public variables are collected to form one

vector variable $y = (y_1, y_2, y_3)^T \in R^8$. The net vector $z = (z_1, z_2, z_3, z_4)^T \in R^4$ gives the common variable values on nets. The partition of $E = [E_1^T, E_2^T, E_3^T]^T \in R^{8\times 4}$ denotes the block of $E$ associated with each subsystem where $E_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$,

$$E_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, E_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, E = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}^T .$$

Our problem shown below finally fits in the standard form.

$$\begin{aligned} min. \quad & f_{B1}(x_1, y_1) + f_{B3}(x_2, y_2) + f_{A2}(x_3, y_3) \\ s.t. \quad & (x_i, y_i) \in C_i, \quad i = 1, 2, 3 \\ & y_i = E_i z, \quad i = 1, 2, 3 \end{aligned} \tag{4.24}$$

Based on this form, the original problem is settled by solving master problem and subproblems alternately. In this paper, we use subgradient method and interior point method to solve master problem and subproblems respectively. In this way, each controller tackles part of problem while trying to reach overall objectives together with coordination.

## System Analysis with Decentralized Model

As an advantage of distributed control, not all information about the system need be collected by main controllers beforehand. So in this section specific information known ahead by the main controllers PCM-B1, PCM-B3, and PCM-A2 is defined. Only with that information are subproblem equations able to be constructed. These parameters are summarized in Table 4.1 and illustrated in Figure 4.3.

Table 4.1: Information needed beforehand

| | **Current Value Needed Beforehand** | **Resistance Value Needed Beforehand** |
|---|---|---|
| PCM-B1 | $i_{Load1}$,$i_{D1}$ | $R_{B1}$,$R_{B2}$,$R_{L1}$,$R_{D1}$ |
| PCM-B3 | $i_{Load2}$, | $R_{B3}$,$R_{B4}$,$R_{L2}$ |
| PCM-A2 | None | $R_{A1}$,$R_{A2}$,$R_{T1}$ |

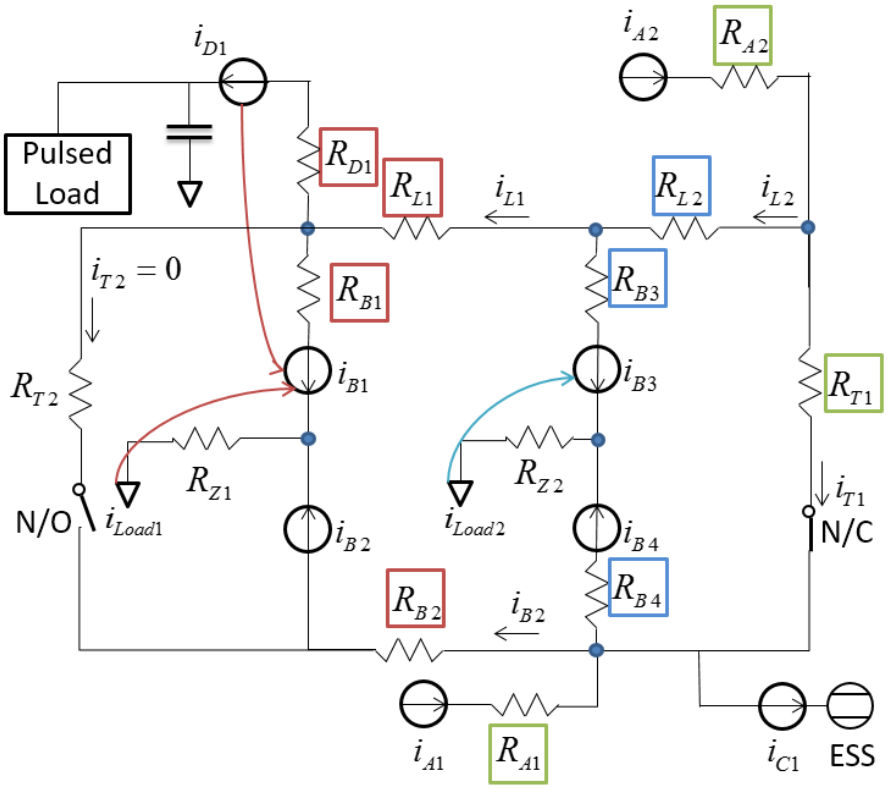


Figure 4.3: Communication about pre-information

As shown in Figure 4.3, information marked in red goes to PCM-B1, blue goes to PCM-B3, and green goes to PCM-A2. All information is collected and used in problem formulation. After this step, during the process of problem solving, main controllers PCM-B1, PCM-B3, and PCM-A2 share and exchange coupling variables to assist each other to converge.

The whole problem is separated into several subproblems. However, the subproblems are not completely independent of each other as the existence of coupling

variables. In each iteration, three subproblems are solved in parallel to evaluate their local optimal value. During this process the coupling variable is fixed. That allows the subproblems to be solved independently. After this inner loop of local optimization, master problem constructs an average value with which the coupling variables are updated. It has been proven that this average moves in the direction to improve the values globally [6].

## Flow Chart

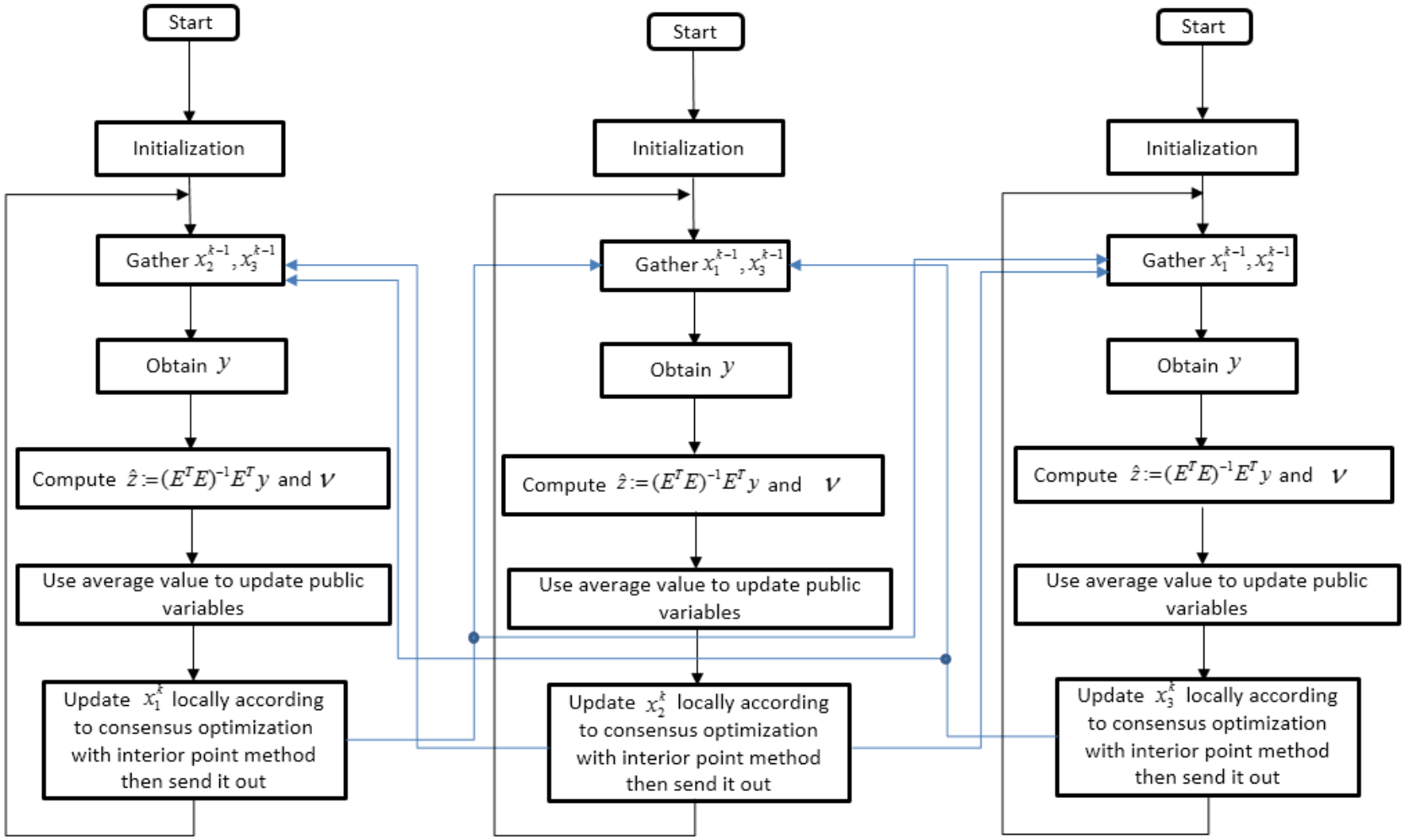


Figure 4.4: System working flow

The Figure 4.4 illustrates the structure of the program to solve the problem in decentralized manner. The black arrows indicate program progress and the blue lines show the information transmission between subprograms. In each inner loop, interior point method is applied to achieve local optimization. In the outer loop, ADMM and dual decomposition method is combined together. Besides, the program is advanced by using average value to update public variables in each outer iteration. The average value behaves better globally than the local optimized value. By using such a better value as a fresh start point for next iteration promises a faster convergence in total.

# Chapter 5

# Results and Convergence

In order to demonstrate the feasibility of the algorithm proposed in Chapter 4, it is evaluated on the shipboard DC microgrid test case for 15s on a detailed model. Then the test case is extended to 50s on a simplified average model which only contains the basic electrical circuit abstraction of the shipboard microgrid. By stripping away the time consuming parts of simulation, the test maintains relevant attributes from the optimization algorithm perspective and allows more cases to be run. Finally the convergence test was fulfilled for three shipboard power models with 3, 5, and 10 zonal loads respectively.

## 5.1 Simulation Results

### Decentralized detailed test

A 15s test case is applied to the shipboard power model in Matlab Simulink. In the beginning, the energy storage system is set up to charge itself, so SOC increases from initial value 0.6 to upper bound 0.7. The decomposition algorithm doesn't apply until 1s. Then at 2.5s, the pulsed load is enabled. Once the pulsed load is detected, the distributed system control enforces the ESS to discharge to accommodate this load until the SOC reaches its lower bound. At 9s, the Zone1 load is decreased to half of its original value and the Zone2 is reduced to one quarter. At 12s, the Zone2 load changes from one quarter to three fourths. The results are shown from Figure 5.1 to Figure 5.5.

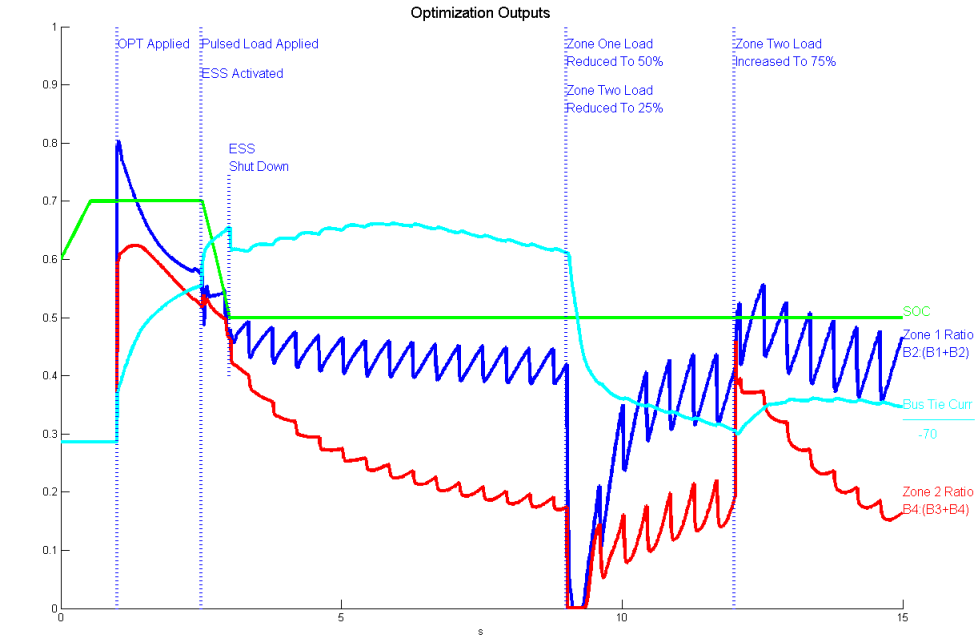


Figure 5.1: Commands in 15s case from decentralized system-level control in the form of bus tie current and sharing ratios of two zones

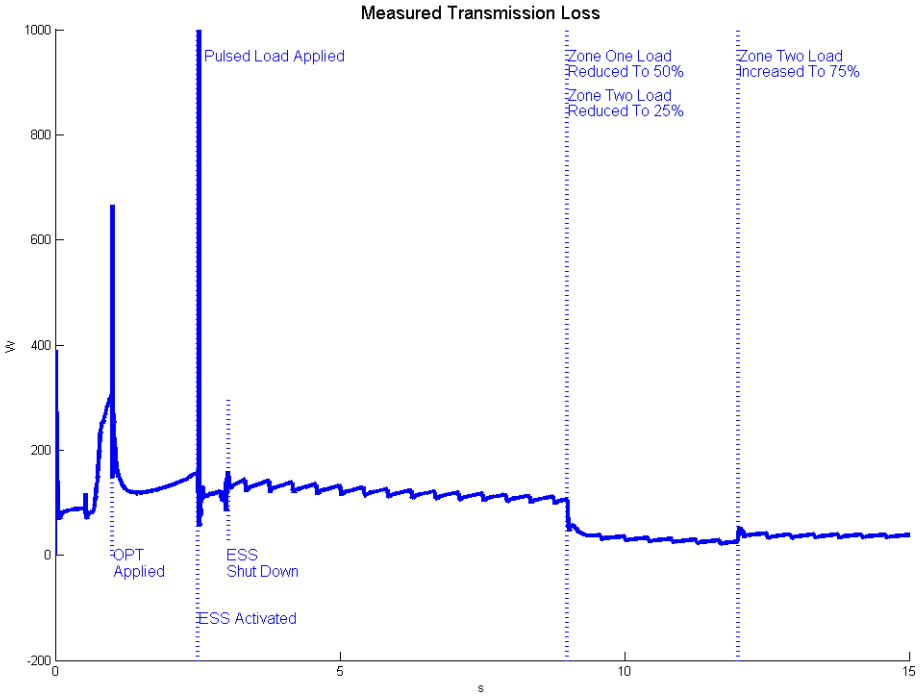


Figure 5.2: Losses in 15s case with decentralized optimization

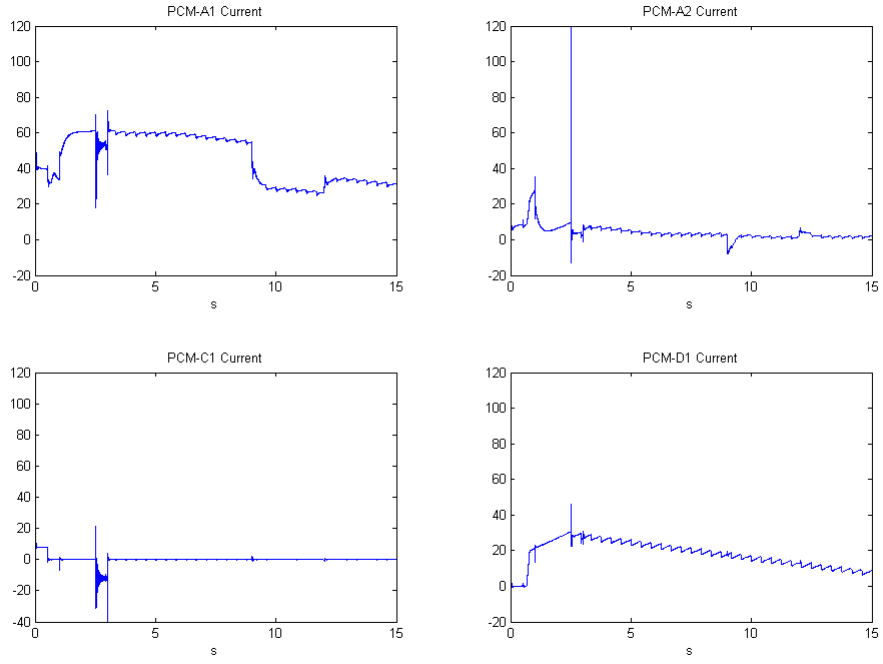


Figure 5.3: Current flows on bus connected converters in 15s case under decentralized system-level control

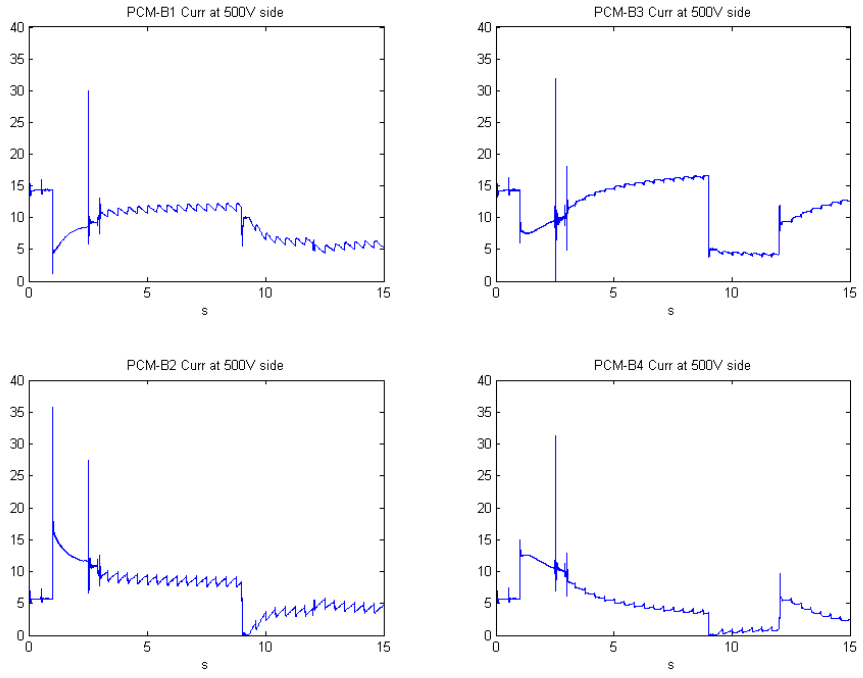


Figure 5.4: Current flows on zonal converters in 15s case under decentralized system-level control

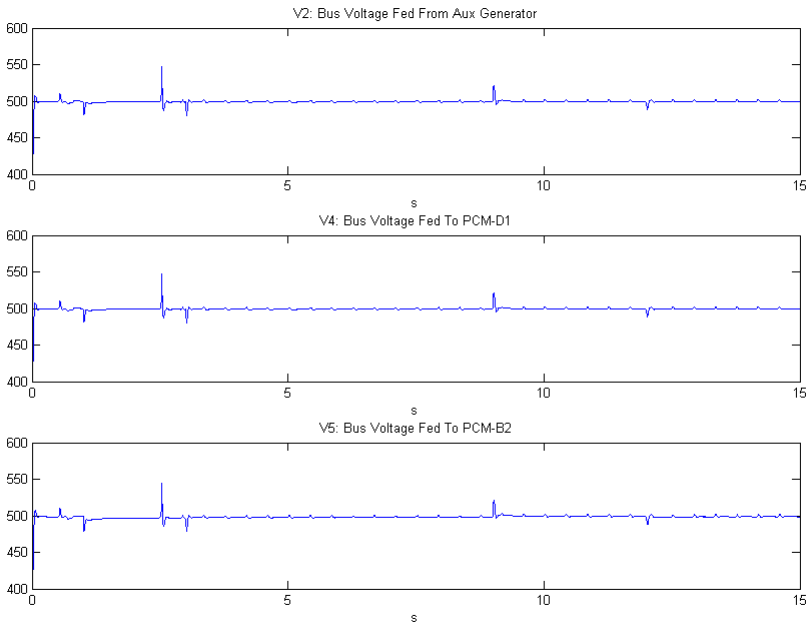


Figure 5.5: Measured bus voltages in 15s case under decentralized system-level control

This test setup is similar to the test case of centralized algorithm. It can be noticed that the decentralized method takes a process to converge to global optimal value. Thus after measuring the change of circuit states, controllers start to operate to reach the new global optimum but it requires several iterations.

## Decentralized 50s simplified test

To better show the trend of the convergence of the proposed distributed algorithm, another test lasting 50s is completed with a simplified model of shipboard DC system. This model has the same outputs as the shipboard model used in 15s centralized case. In other words, the optimization algorithm is handling the same input data as the not-simplified test system.

In the test setup, optimization is applied at 10s, pulsed load is not active until 25s, and the changes of loads happen at 35s and 45s respectively. With the similar test environment as the centralized case, the converged data obtained are directly comparable. 10-25s here corresponds to 1-2.5s in centralized 15s case. Similarly 25-35s, 35-45s, and 45-50s here correspond to 2.5-9s, 9-12s, and 12-15s respectively in the 15s centralized test case. Figure 5.6 depicts SOC, bus tie current, and load sharing ratios in two zones. By comparing it with Figure 3.1 , it can be seen that all steady values are very close. For instance, the sharing ratio for Zone2 is converged to about 0.5 from Figure 3.1 when load in Zone1 is decreased to a half and load in Zone2 is reduced to one quarter. This value can also be reached in the decentralized control as shown in Figure 5.6. Figure 5.7 depicts the losses during the process and Figure 3.2 is very similar to it. Figure 5.8 and Figure 5.9 are corresponding to Figure 3.3 and Figure 3.4 respectively.

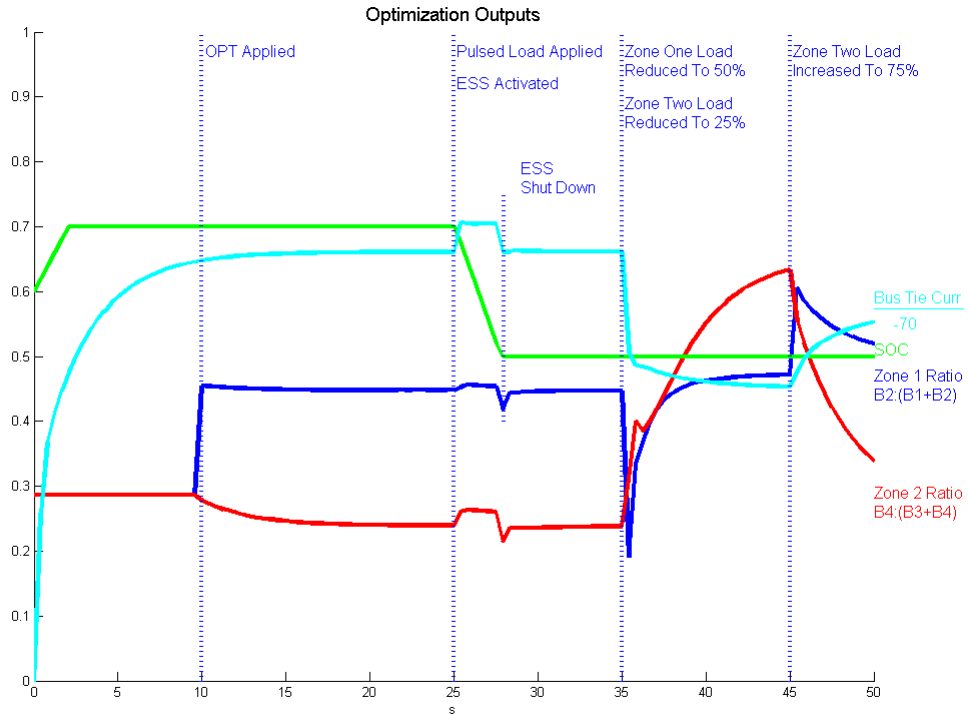


Figure 5.6: Commands in 50s case from decentralized system-level control in the form of bus tie current and sharing ratios of two zones

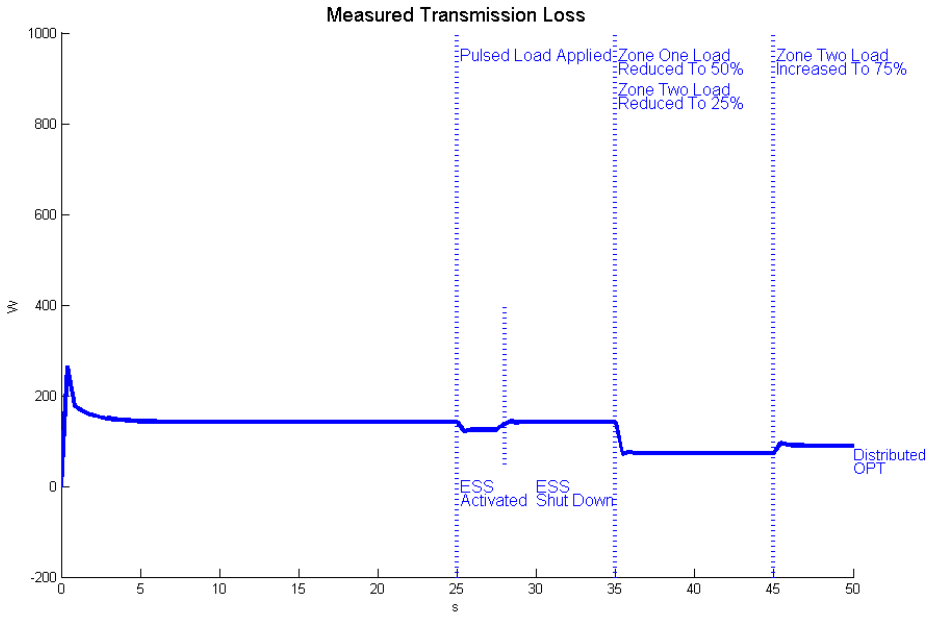


Figure 5.7: Losses in 50s case with decentralized optimization

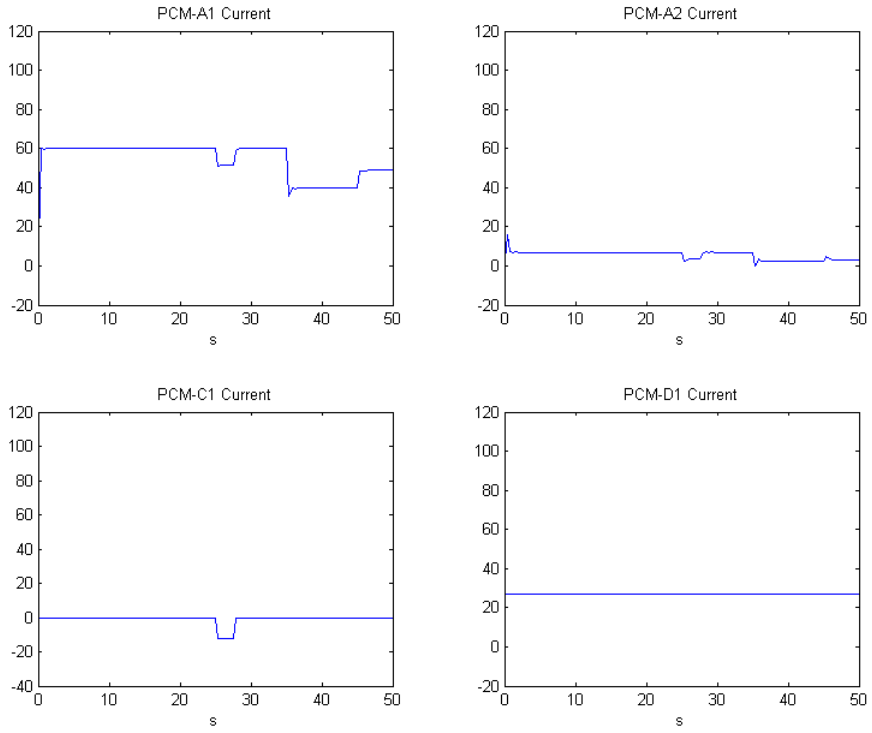


Figure 5.8: Current flows on bus connected converters under 50s decentralized system-level control

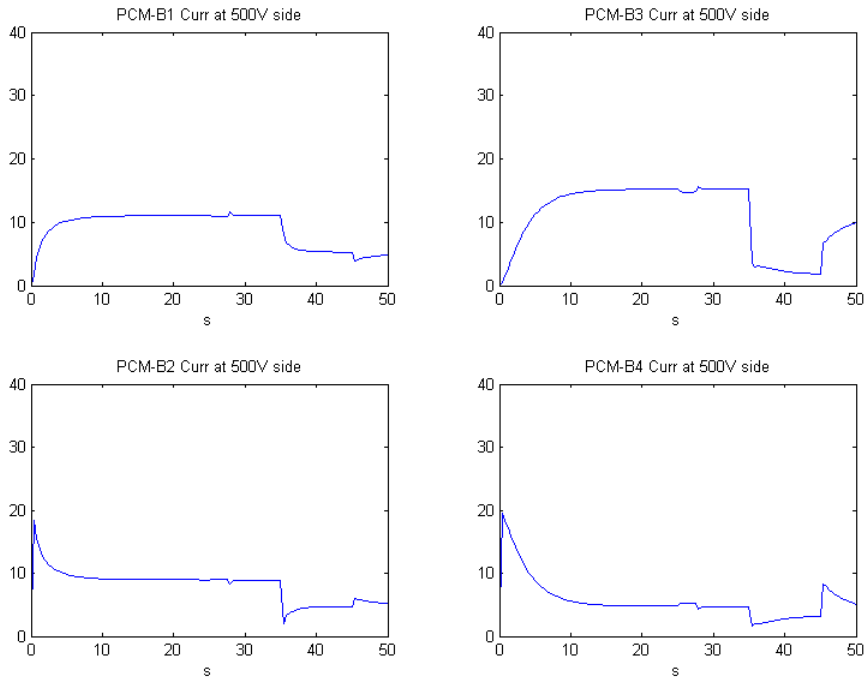


Figure 5.9: Current flows on zone converters under 50s decentralized system-level control

## Numerical Results Comparison

To illustrate that all results from the centralized and decentralized methods are consistant, Table 5.10 lists the numerical results for each load status in 15s centralized and 50s decentralized cases.

| | State 1 | | State 2 | | State 3 | | State 4 | |
|---|---|---|---|---|---|---|---|---|
| | Centralized | Distributed | Centralized | Distributed | Centralized | Distributed | Centralized | Distributed |
| IA1(p.u.) | 1.99999 | 1.9999 | 1.721 | 1.7209 | 1.32077 | 1.3207 | 1.63449 | 1.6343 |
| IA2(p.u.) | 0.23334 | 0.2335 | 0.112359 | 0.1125 | 0.0792324 | 0.0793 | 0.0988482 | 0.0989 |
| IC1 (p.u.) | 0.00E+00 | 0 | -0.399976 | -0.4 | 0 | 0 | 0 | 0 |
| ID1 (p.u.) | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 |
| IB1(p.u.) | 0.368455 | 0.3684 | 0.365122 | 0.3651 | 0.175387 | 0.1754 | 0.166748 | 0.1667 |
| IB2(p.u.) | 2.98E-01 | 0.2982 | 3.02E-01 | 0.3015 | 1.58E-01 | 0.158 | 1.67E-01 | 0.1666 |
| IB3(p.u.) | 5.08E-01 | 0.5072 | 4.97E-01 | 0.4965 | 5.81E-02 | 0.058 | 3.63E-01 | 0.3626 |
| IB4(p.u.) | 1.59E-01 | 0.1595 | 1.70E-01 | 0.1702 | 1.09E-01 | 0.1087 | 1.37E-01 | 0.1375 |
| IT1(p.u.) | -1.54271 | -1.5422 | -1.64936 | -1.6491 | -1.05422 | -1.0541 | -1.3307 | -1.3303 |
| ratio1 | 4.47E-01 | 0.4473447 | 4.52E-01 | 0.45229523 | 4.74E-01 | 0.47390522 | 5.00E-01 | 0.49984998 |
| ratio2 | 0.2386007 | 0.239238 | 0.25511087 | 0.255287236 | 0.65162987 | 0.65206959 | 0.274404 | 0.27494501 |
| Loss(p.u.) | 0.0094299 | 0.009431 | 0.00828595 | 0.008286 | 0.00484598 | 0.004846 | 0.00591404 | 0.005914 |

Figure 5.10: Numerical results for studied cases

## Result Analysis

### Step size

Step size is one of the key parameters that can affect system convergence. In the experiences above, a fixed step size is shared by all main controllers. Dynamic step size can also be applied in this decomposition frame but it is shown that dynamic step size does not provide many benefits. The following four figures are collected with step size setup to $\frac{0.01}{k}$, $\frac{0.1}{k}$, $\frac{0.01}{\sqrt{k}}$, and $\frac{0.1}{\sqrt{k}}$.

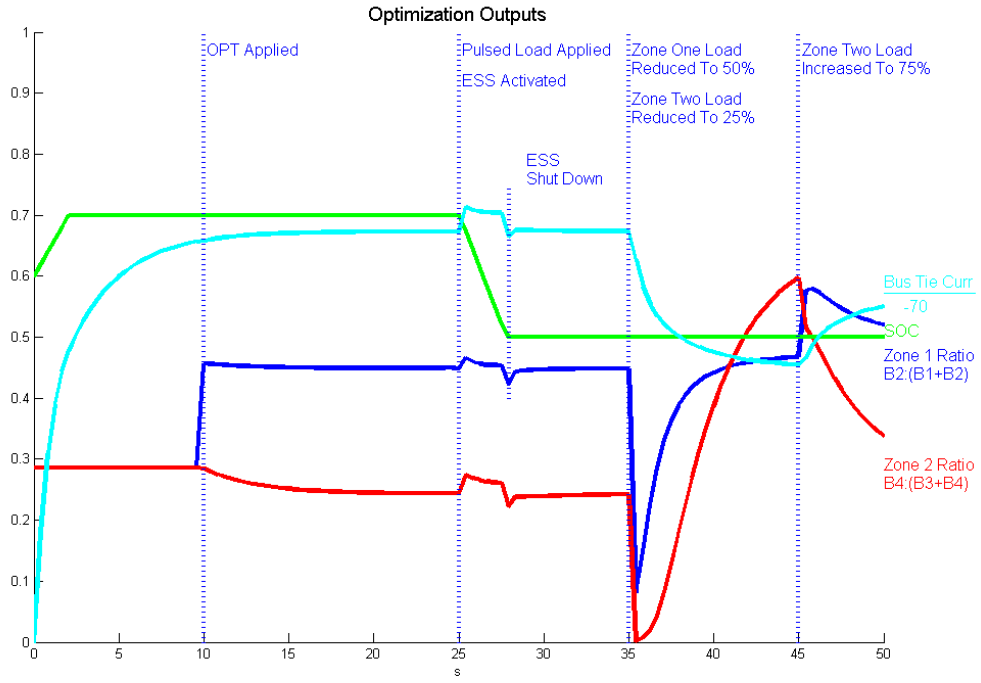


Figure 5.11: Commands with step size set to $\frac{0.01}{k}$

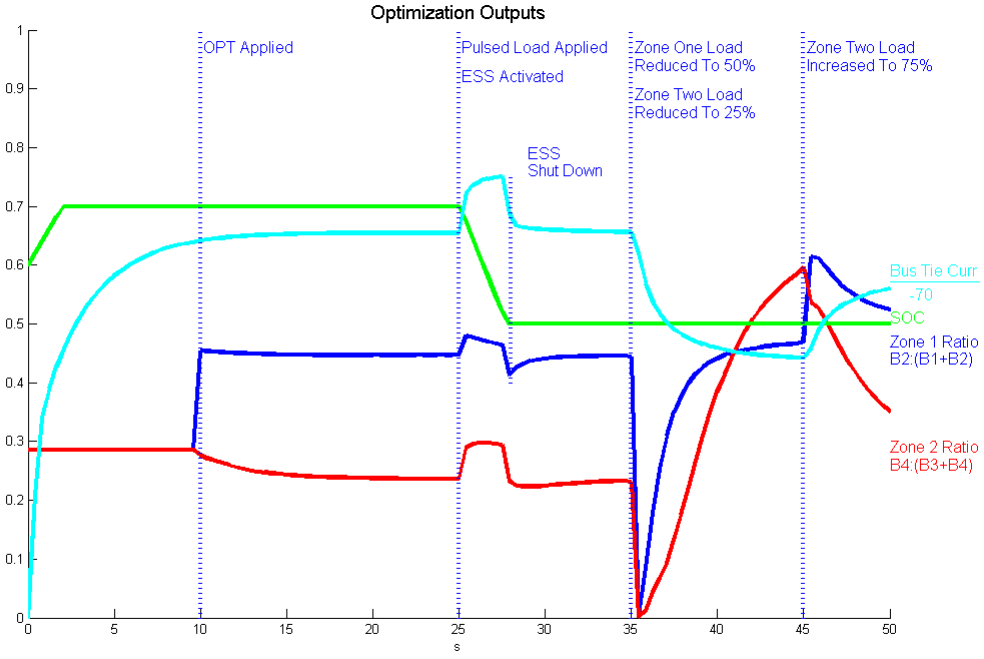


Figure 5.12: Commands with step size set to $\frac{0.1}{k}$

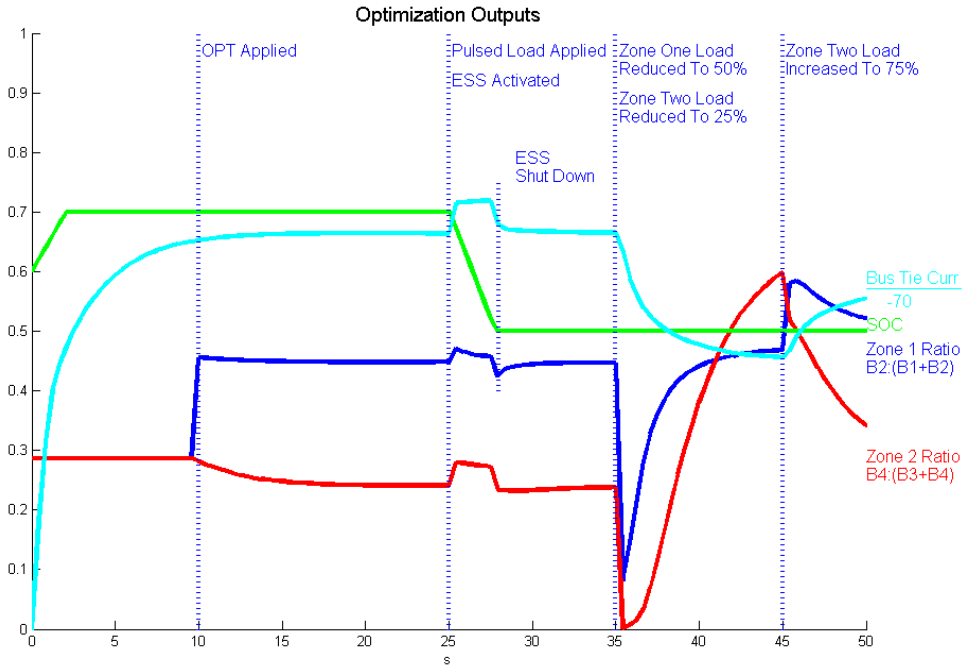


Figure 5.13: Commands with step size set to $\frac{0.01}{\sqrt{k}}$

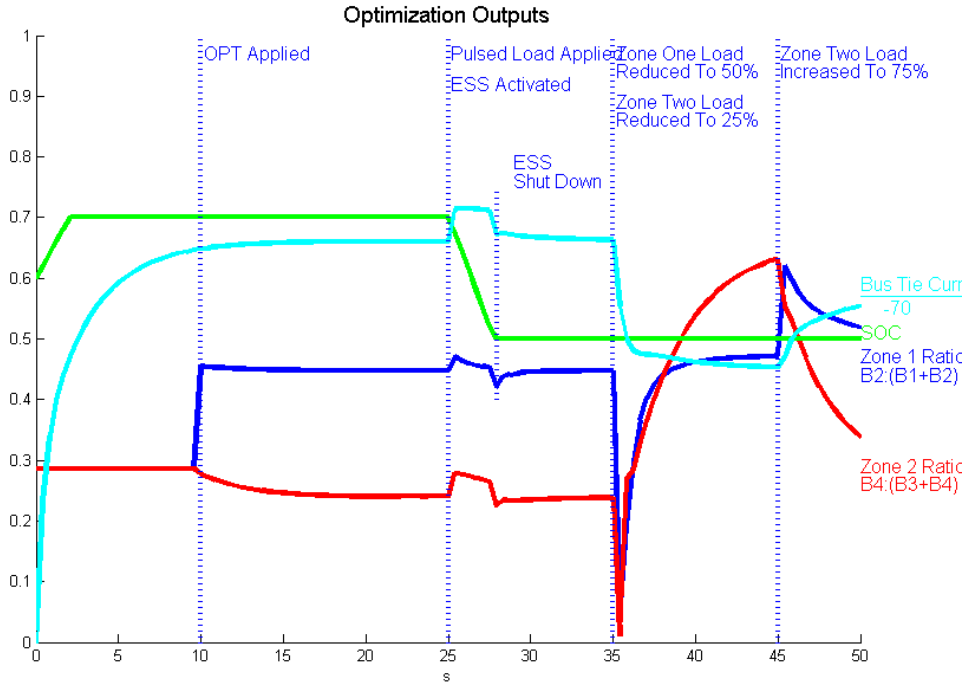


Figure 5.14: Commands with step size set to $\frac{0.1}{\sqrt{k}}$

One disadvantage of the dynamic step size is that it requires a reset to all main controllers wherever change happens. When a sudden change happens in one subsystem, a local recalculation would occur first. Then all main controllers share the same step length to reach global convergent result. This action enforces main controllers in other subsystems being aware of the change. This sharing conflicts with the concept of decomposition method which is each local main controller only cares its own local changes.

Tests merging both fixed and dynamic step sizes are also considered. Figure 5.15 shows the case with step size of 0.2 for first 6 iteration after change then 0.1 ever. By comparing it with Fiugre 5.6, it is noticed that 0.2 provides quicker transient convergent process than 0.1. So the procedure value of step size has influence on the transient performance.

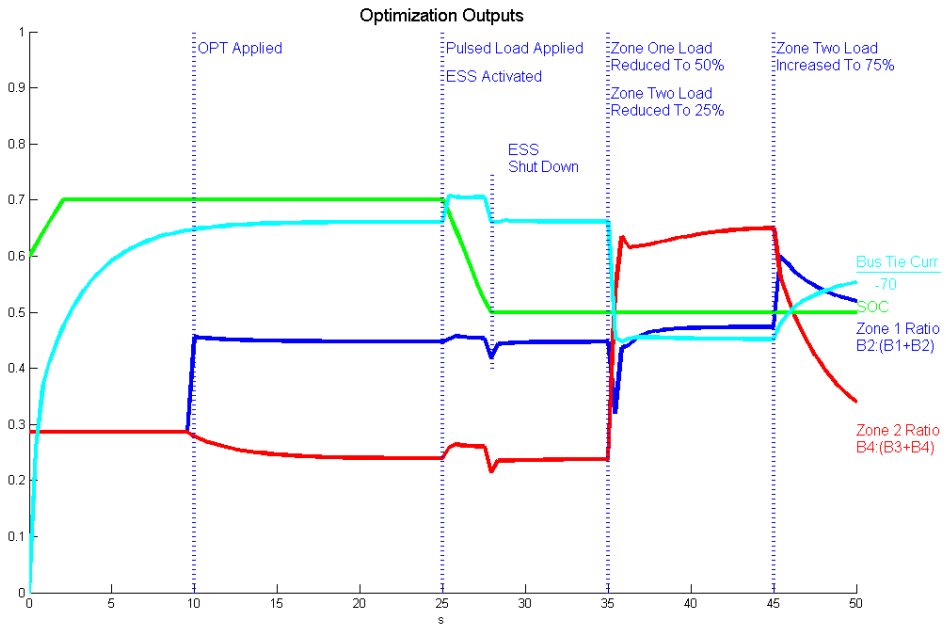


Figure 5.15: Commands with step size set to 0.2 then to 0.1

In this work, error and trial is implied to pick up a proper step size. The figures showed ahead in the 50s and 15s tests are obtained from a step size of 0.1. The

system turns out to be not convergent with this value set to 1. Figure 5.16 shows control commands when this step size is 0.01.

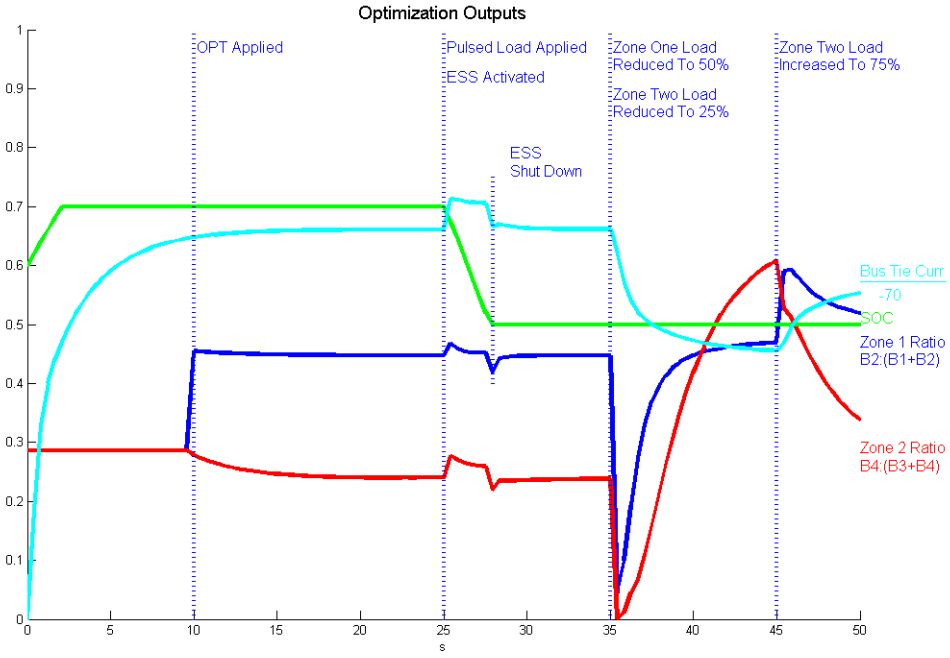


Figure 5.16: Commands with step size set to 0.01

### Transition

By comparing Figure 5.16 with Figure 5.6, we notice that in Figure 5.16 the sharing ratios going down to zero after load changes such as those occurring at 36s. These drops are caused by the nature of decomposition optimization. Take Zone2 as an example because it is the worst case compared to Zone1. This sharing ratio is defined as current through PCM_B4 over the load in Zone2. It is calculated by main controller PCM_B4. At the moment of load decreasing, PCM_B4 who is monitoring $i_{Load2}$ inspects the changes, but this influence from its adjacent systems is low for this moment. PCM_B4 isn't aware of the load in Zone1 has changed too. So in the very beginning time, the commands are for local optimum. Because the resistance in distribution line from PCM_B3 side is 0 and the resistance from PCM_B4 side

is 0.097, local optimization orders the power through PCM_B4 drop to about zero. Then gradually the influence from outside is coming in through coupling variables, this sharing ratio starts to converge to global optimum through iteration. So this value is bounced back to the optimum value after the drop.

The reason why there is no obvious drop in Figure 5.6 is because the step size there is bigger. With a bigger step size, the effect from coupling variables is enhanced, so the convergent trend is more obvious in the beginning of iteration. If we over-adjust this number, overshoot can be observed at the instance of a change. For instance, Figure 5.17 shows the step size as 0.8 for initial 6 steps then back to 0.1 further.

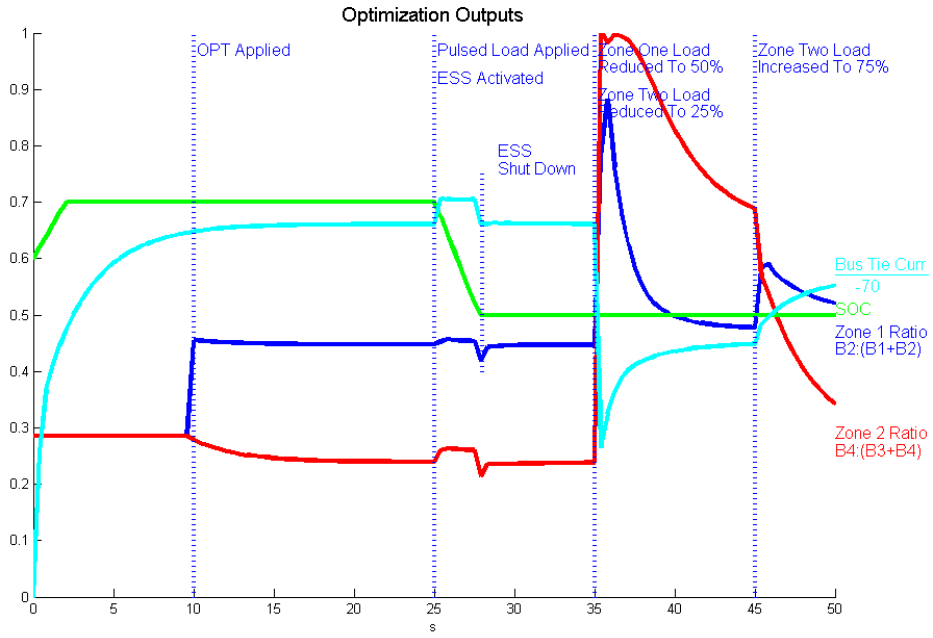


Figure 5.17: Commands with step size set to 0.8 then to 0.1

When fixed step size is used, the system has the best transient performance before a specific value. Figure 5.18 indicates result with the step size of 0.8. The result here is better than step size of 0.1 as shown in Figure 5.6. However, any value higher than this will lead the system to un-convergent. In this shipboard microgrid model, the system is not able to converge when step size is bigger than 0.9.

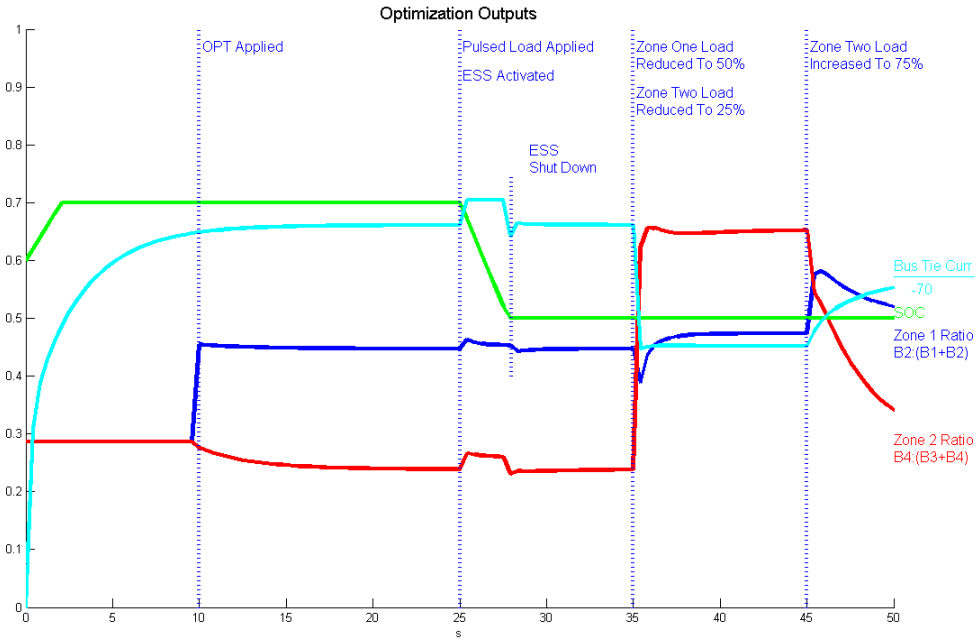


Figure 5.18: Commands with step size set to 0.8

### Sampling Time

The sampling time used in the previous simulations is 20ms. By changing this time to 10ms, we reach a new set of commands as shown in Figure 5.19. Here the convergence speeds up because convergence time is proportional to sampling time if iteration number is fixed. However, the computation requirements increase since all iterations must complete in a shorten time frame.

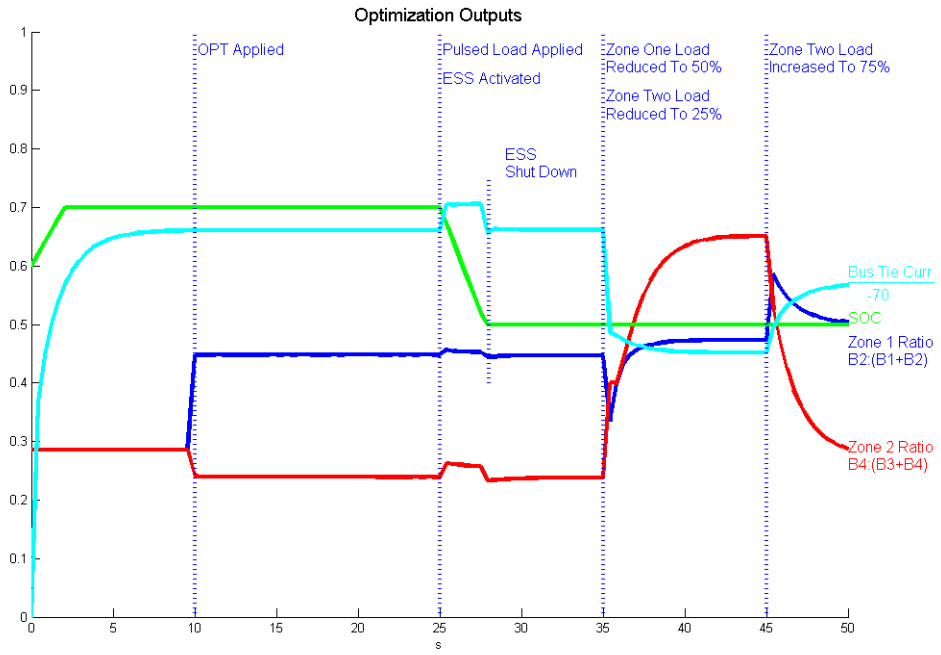


Figure 5.19: Commands under 10ms sampling time

## 5.2 Convergence analysis

In order to show the convergence behavior of the proposed decomposition algorithm, it is applied to two more microgrid systems with five and ten load-zones respectively. The status of loads applied to the system at 35s to 45s in the 50s test case illustrated before is applied on all these three microgrids. Difference of system loss and difference of dual variables are plotted versus iteration number. Figure 5.20 depicts the evolution of these variables for the studied system with two zones. This system is partitioned as shown in Figure 4.1. In Figure 5.20, we plot three lines to indicate that dual variables decreased 70%, 95%, and 99% of its initial value. The according iteration numbers are also listed in the same figure. It takes 9 iterations to drop to 70%, 61 more iterations to 95%, and 74 more iterations to 99%.

The system with five zones is separated into six subsystems as shown in Figure 5.21 , one is about the generators and the other five are related to five load-zones respectively. Figure 5.22 shows that the convergence speed is slower comparing to

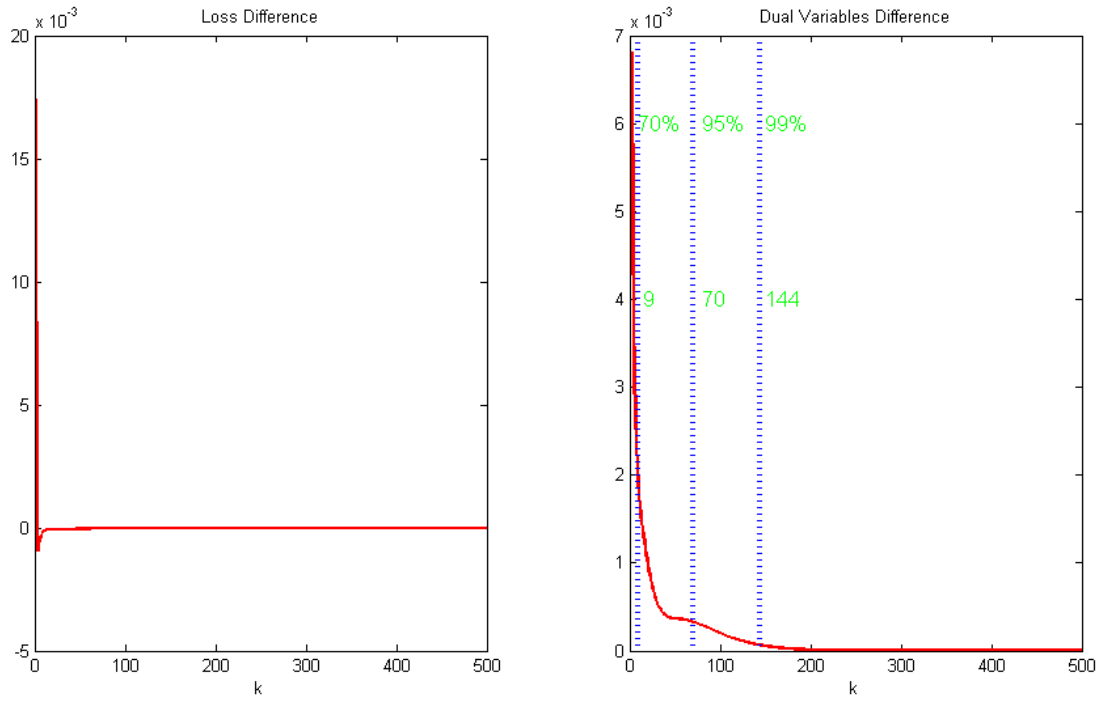


Figure 5.20: Iterations in System with Two Zones

the system with two zones. It takes 12 iterations to drop to 70%, 88 iterations from 70% to 95%, and 134 iterations from 95% to 99%.

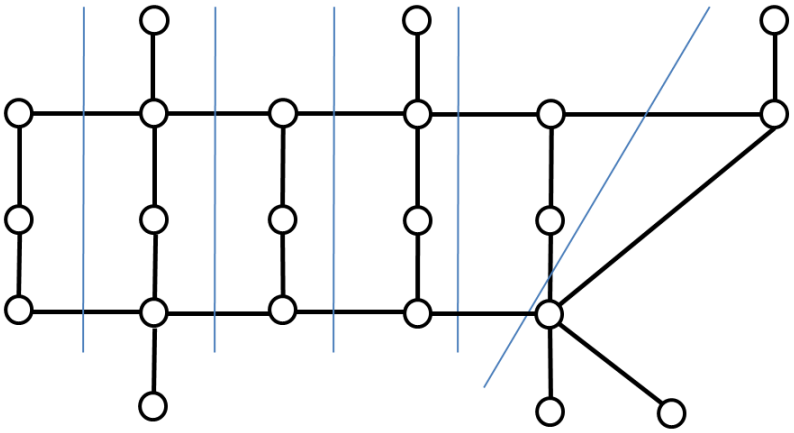

Figure 5.21: Partition in System with Five Zones

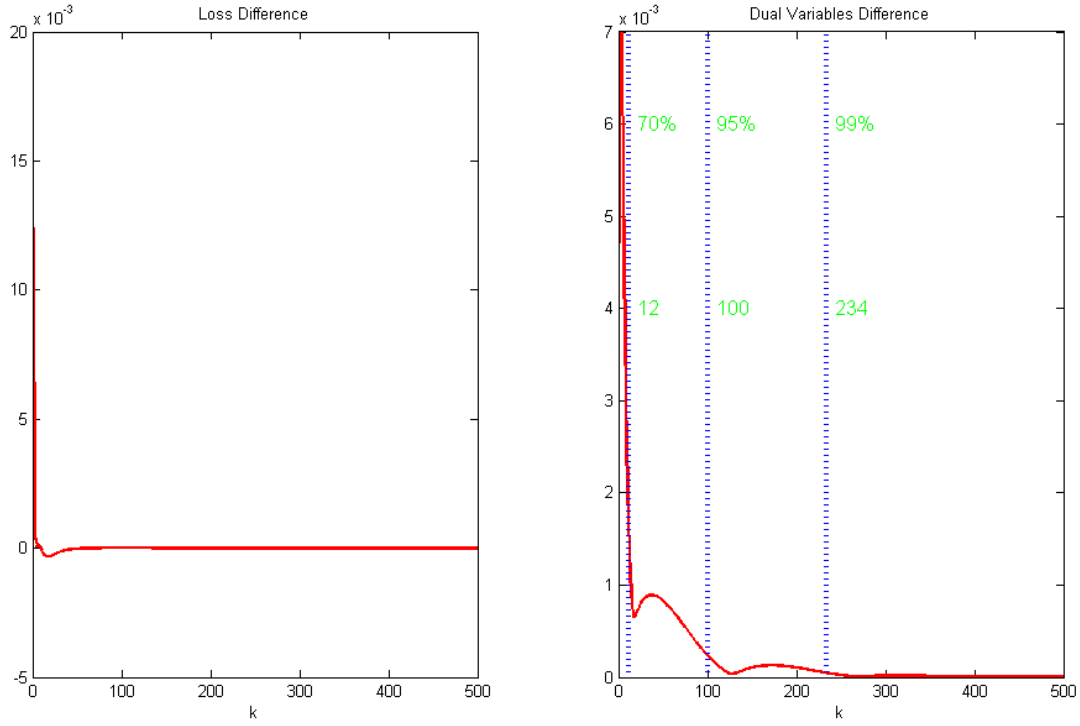


Figure 5.22: Iterations in System with Five Zones

Figure 5.23 shows the partition of the system with 10 load-zones. One subsystem for the generators plus ten for the load-zones makes eleven subsystems in total. They only communicates with adjacent systems. When there is some changes happening in the far left zone, it firstly spreads out to its neighbor on the right hand with one iteration, then to neighbor's neighbor and gradually propagate to the whole system. So the generator zone in the far right could not be aware of the load change in the first few iterations. That's why this system takes more iterations to converge. Figure 5.24 shows the convergence evolution. It takes 21 iterations for the difference of dual variables to drop to 70%, 155 iterations from 70% to 95%, and 188 iterations from 95% to 99%.

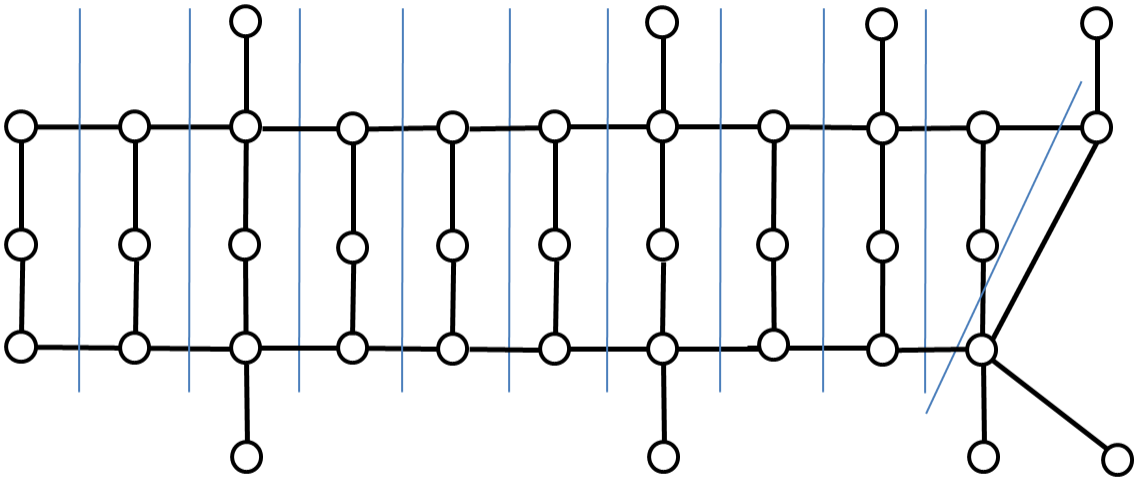

Figure 5.23: Partitions in System with Ten Zones

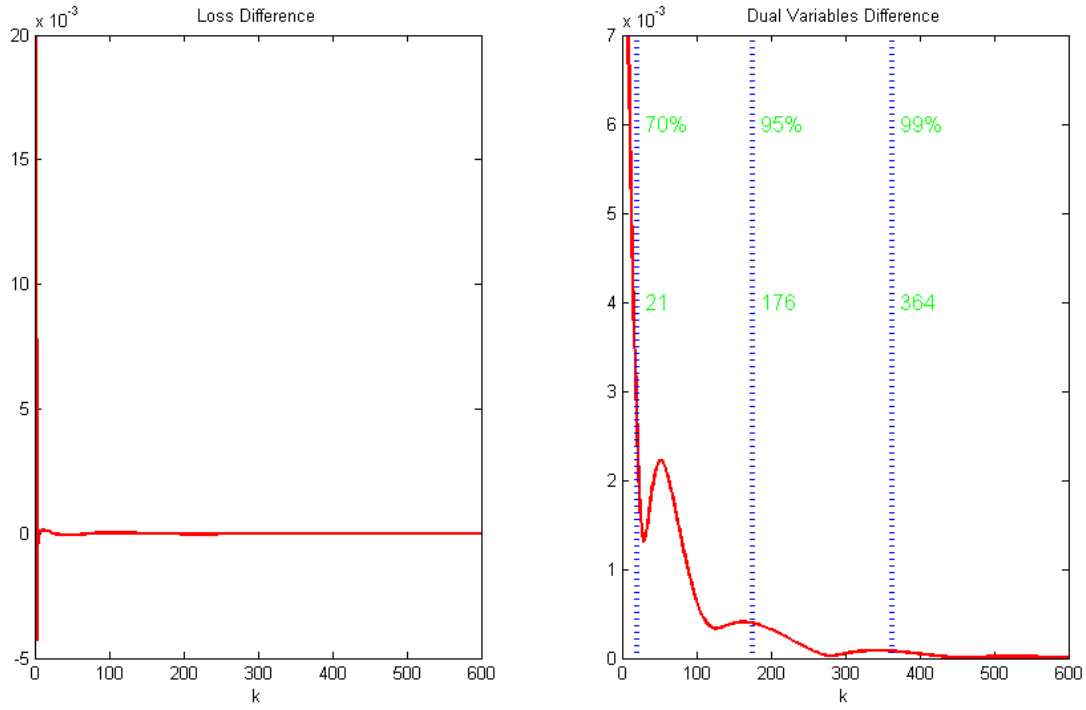


Figure 5.24: Iterations in System with Ten Zones

The above analysis is about the process of convergence which is one of the most obvious differences of the decentralized control from the centralized control. Due to the evolution of convergence, decentralized control is slower than centralized control. To reach the same performance as centralized control, decentralized method needs higher sampling rate. However, the decentralized method immunes to one points of failure by its nature. And each controller requires only local information and deals with relatively small problems comparing to centralized methods. The local problem size depends on the partition of the system. By introducing more subsystems, the problem could be split into more local problems which run in parallel and work coordinately to reach global optimum.

# Chapter 6

# Conclusion

DC microgrid power systems typically include various power electronics devices which can be coordinated by system-level control to achieve system level goals such as efficient energy distribution. This requires a framework to determine how to perform coordination to achieve a given system level goal.

Managing energy distribution to minimize costs is achieved through system level optimization. In this dissertation, the interior point method is deployed on the shipboard DC microgird to calculate an optimal coordination solution centrally. The centralized method could achieve an optimal solution within a sufficiently small timeframe. For instance, 45ms is used as the sampling period in an example shipboard DC microgrid with centralized optimization method. However a centralized framework is vulnerable to single points of failure as well as not being easily scalable.

To address these concerns, a distributed optimization method is proposed and implemented. In this framework, the entire power grid is split into several subgrids with the interior point algorithm applied to achieve local optimization individually. The value of coupled variables across the subgrids are then averaged and pushed back to each local optimization problem under the dual decomposition and ADMM framework. In this way, variables would eventually reach a global optimal across the entire grid.

Although both methods reach the same optimal results, they naturally differ from each other. With the centralized method, only one single iteration is needed to update orders to lower level control. However, for the decomposition method, it

takes steps to propagate local information and update public variables around the grid. For instance, it requires 70 iterations to reach 95% accuracy in the case of the two load zone test system which is divided into three subsystems. Moreover, in the decentralized system the communication is localized within subsystems. Local main controllers handle local changes rather than a remote central controller. Only public variables are transmitted between relevant main controllers. Furthermore, during the iteration of the decomposition method, the objective function value reduces faster at the early stage compared to when it is approaching the optimal. As shown in Chapter 5, for the example microgrids with 2, 5, and 10 zones, 70% accuracy is achieved within 9, 12, and 21 steps respectively as compared to 144, 234, and 364 steps for 99% accuracy. So the decomposition method can be used to find a solution within a few percent of the optimum value in a reasonable period of time instead of cases that very accurate results are required.

The simulation results also show how other factors affect the performance of the decomposition method. Firstly we consider the effects of different sampling rates. From the point of decomposition algorithm itself, higher sampling rate speeds up convergence as shown by comparing Figure 5.6 and Figure 5.19. Secondly, simulation experiments conducted with dynamically adjusted step size demonstrated that transient performance is not improved much. Furthermore, because dynamic step size requires the sharing of information in the entire system, it increases the information that must be shared. Experiments with different fixed step length are also conducted. The decomposition algorithm keeps convergent if the step size is under 0.8 in this specific case. Although higher value leads to unstable system, the value as closer to 0.8 provides better performance during system state transitions. Thus, the selection of step size is an important factor and should in general be as large as possible for stable operation.

The objective function for the test shipboard microgrid doesn't involve the gen-

erator loss, so unit commitment problem is not incorporated. The unit commitment problem deals with on/off schedule of the generators set. More constraints including spinning reserve or ramp-up time need to be considered. These can be added into future work.

# Bibliography


[1] Dimitrios Athanasiadis and Stephen McArthur, *Active network management using distributed constraint optimisation*, 2013 IEEE Power & Energy Society General Meeting, IEEE, 2013, pp. 1–5.

[2] Rahmat Azami, Mohammad Sadegh Javadi, and Ghasem Hematipour, *Economic load dispatch and dc-optimal power flow problem-pso versus lr*, Int. J. Multidscip. Sci. Eng **2** (2011), no. 9, 8–13.

[3] Faycal Bouhafs and Michael Mackay, *Active control and power flow routing in the smart grid*, Newsletter (2012).

[4] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein, *Distributed optimization and statistical learning via the alternating direction method of multipliers*, Foundations and Trends® in Machine Learning **3** (2011), no. 1, 1–122.

[5] Stephen Boyd and Lieven Vandenberghe, *Convex optimization*, Cambridge university press, 2004.

[6] Stephen Boyd, Lin Xiao, Almir Mutapcic, and Jacob Mattingley, *Notes on decomposition methods*, Notes for EE364B, Stanford University (2007), 1–36.

[7] KL Butler and M Ehsani, *Flexible ship electric power system design*, Proc of the Symposium of Engineering the Total Ship, 1998.

[8] EM Davidson, MJ Dolan, SDJ McArthur, and GW Ault, *The use of constraint programming for the autonomous management of power flows*, Intelligent System Applications to Power Systems, 2009. ISAP'09. 15th International Conference on, IEEE, 2009, pp. 1–7.

[9] Shilpa B Ganesh, *Multiagent autonomous energy management*, Ph.D. thesis, West Virginia University, 2005.

[10] SY Ge and TS Chung, *Optimal active power flow incorporating power flow control*

*needs in flexible ac transmission systems*, IEEE Transactions on Power Systems **14** (1999), no. 2, 738–744.

[11] H.L. Ginn, *Ginn onr program review*, Office of Naval Research, 2012.

[12] Ibrahim Abou Hamad, Per Arne Rikvold, and Svetlana V Poroseva, *Floridian high-voltage power-grid network partitioning and cluster optimization using simulated annealing*, Physics Procedia **15** (2011), 2–6.

[13] Nikos Hatziargyriou, *Microgrids: architectures and control*, John Wiley & Sons, 2013.

[14] Rabih A Jabr, *Modeling network losses using quadratic cones*, IEEE Transactions on Power Systems **20** (2005), no. 1, 505–506.

[15] Björn Johansson, Pablo Soldati, and Mikael Johansson, *Mathematical decomposition techniques for distributed cross-layer optimization of data networks*, IEEE Journal on Selected Areas in Communications **24** (2006), no. 8, 1535–1547.

[16] Narendra Karmarkar, *A new polynomial-time algorithm for linear programming*, Proceedings of the sixteenth annual ACM symposium on Theory of computing, ACM, 1984, pp. 302–311.

[17] Farid Katiraei, Reza Iravani, Nikos Hatziargyriou, and Aris Dimeas, *Microgrids management*, IEEE Power and Energy Magazine **6** (2008), no. 3, 54–65.

[18] Faridaddin Katiraei and Mohammad Reza Iravani, *Power management strategies for a microgrid with multiple distributed generation units*, IEEE transactions on power systems **21** (2006), no. 4, 1821–1831.

[19] Benjamin Kroposki, Robert Lasseter, Toshifumi Ise, Satoshi Morozumi, Stavros Papathanassiou, and Nikos Hatziargyriou, *Making microgrids work*, IEEE Power and Energy Magazine **6** (2008), no. 3, 40–53.

[20] R Lasseter, A Akhil, C Marnay, J Stevens, J Dagle, R Guttromson, AS Meliopoulous, R Yinger, and J Eto, *White paper on integration of distributed energy resourcesâĂŤthe microgrid concept*, Consortium for Electric Reliability Technology Solutions (CERTS) April (2002).

[21] Robert H Lasseter, *Microgrids*, Power Engineering Society Winter Meeting, 2002. IEEE, vol. 1, IEEE, 2002, pp. 305–308.

[22] Yunwei Li, D Mahinda Vilathgamuwa, and Poh Chiang Loh, *Design, analysis, and real-time testing of a controller for multibus microgrid system*, IEEE Transactions on Power Electronics **19** (2004), no. 5, 1195–1204.

[23] Xiong Liu, Peng Wang, and Poh Chiang Loh, *A hybrid ac/dc microgrid and its coordination control*, IEEE Transactions on Smart Grid **2** (2011), no. 2, 278–286.

[24] Xin Lou and Chee Wei Tan, *Convex relaxation and decomposition in large resistive power networks with energy storage*, Smart Grid Communications (SmartGridComm), 2013 IEEE International Conference on, IEEE, 2013, pp. 642–647.

[25] Sam Miller, Sarvapali D Ramchurn, and Alex Rogers, *Optimal decentralised dispatch of embedded generation in the smart grid*, Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1, International Foundation for Autonomous Agents and Multiagent Systems, 2012, pp. 281–288.

[26] Pragnesh Jay Modi, Wei-Min Shen, Milind Tambe, and Makoto Yokoo, *An asynchronous complete method for distributed constraint optimization*, AAMAS, vol. 3, 2003, pp. 161–168.

[27] ———, *Adopt: Asynchronous distributed constraint optimization with quality guarantees*, Artificial Intelligence **161** (2005), no. 1, 149–180.

[28] James A Momoh, ME El-Hawary, and Ramababu Adapa, *A review of selected optimal power flow literature to 1993. part i: Nonlinear and quadratic programming approaches*, IEEE transactions on power systems **14** (1999), no. 1, 96–104.

[29] ———, *A review of selected optimal power flow literature to 1993. part ii: Newton, linear programming and interior point methods*, IEEE Transactions on Power Systems **14** (1999), no. 1, 105–111.

[30] The Institution of Engineering and Technology, *What is a smart grid?*, The IET, 2013.

[31] Daniel F Opila and Luke Solomon, *Optimal control of dynamic pulse power loads in naval power systems using the pontryagin minimum principle and dynamic programming*, 2012 IEEE Power and Energy Society General Meeting, IEEE, 2012, pp. 1–7.

[32] Daniel Pérez Palomar and Mung Chiang, *A tutorial on decomposition methods for network utility maximization*, IEEE Journal on Selected Areas in Communications **24** (2006), no. 8, 1439–1451.

[33] Paolo Piagi and Robert H Lasseter, *Autonomous control of microgrids*, 2006 IEEE Power Engineering Society General Meeting, IEEE, 2006, pp. 8–pp.

[34] Florian A Potra and Stephen J Wright, *Interior-point methods*, Journal of Computational and Applied Mathematics **124** (2000), no. 1, 281–302.

[35] Victor H Quintana, Geraldo L Torres, and Jose Medina-Palomo, *Interior-point methods and their applications to power systems: a classification of publications and software codes*, IEEE Transactions on power systems **15** (2000), no. 1, 170–176.

[36] Margaret Wright, *The interior-point revolution in optimization: history, recent developments, and lasting consequences*, Bulletin of the American mathematical society **42** (2005), no. 1, 39–56.

[37] Stephen J Wright, *Primal-dual interior-point methods*, Siam, 1997.

[38] Yinyu Ye, *Interior point algorithms: theory and analysis*, vol. 44, John Wiley & Sons, 2011.

[39] Makoto Yokoo, *Distributed constraint satisfaction: foundations of cooperation in multi-agent systems*, Springer Science & Business Media, 2012.

[40] Ray Daniel Zimmerman, Carlos Edmundo Murillo-Sánchez, and Robert John Thomas, *Matpower: Steady-state operations, planning, and analysis tools for power systems research and education*, IEEE Transactions on power systems **26** (2011), no. 1, 12–19.

# Appendix A

# Matlab Code For Testing The Decentralized Algorithm

Three similar sets of Matlab codes run in parallel for the subsystems. The set in PCM-B1 is shown as follows.

Listing A.1: AgentPCM-B1

```
1
2  function Update(block)
3
4  s = 15000;            %  Base power in P.U. is 15kw
5  vh = 500;             %  Base voltage in P.U. is 500V
6  rh = vh^2/s;          %  Base resistance
7  rD1 = 0.05/rh;        %  Rd1 Resistance in P.U.
8  rB1 = 0.12/rh;        %  Rb1 Resistance in P.U.
9  rB2 = 0.2/rh;         %  Rb2 Resistance in P.U.
10 rL1 = 1e-7/rh;        %  RL1 Resistance in P.U.
11
12 i_load1 = block.InputPort(1).data(1);     %Zone1 load as input
13 i_D1 = block.InputPort(1).data(2);        %Pulsed load as input
14 y1 = block.Dwork(2).data;                 %Variables updates from
      Previous step
15 y1_p = y1(2:3);                           %Public variables updates
```

```
    from itself
y2_p = block.InputPort(3).data ;          %Public variables updates
    from PCM_B3
y3_p = block.InputPort(4).data ;          %Public variables updates
    from PCM_A2

A1 = [1  1  0;                    %Local constraints system matrix
      1  0 -1];
beq1 = [i_load1;-i_D1];           %Local constraints system inputs
lb1=[0;0;-5];                     %Lower bound for variables
ub1=[10/15;10/15;5];              %Higher bound for variables
E = [1 0 0 0;                     %Matrix defined by public variable
    and net vector
     0 0 1 0;
     0 1 0 0;
     0 0 1 0;
     0 0 0 1;
     1 0 0 0;
     0 1 0 0;
     0 0 0 1;];
alpha_k = 0.1;                     %Fixed step size

%Interior point method is used for inner loop.
options = optimset('Algorithm','interior-point', 'Diagnostics', '
   off', 'Display', 'off');

v = block.Dwork(3).Data;           %Dual variables
```

```
38 y = [y1_p; y2_p; y3_p];              %All public variables
39 Zcap = (inv(E' * E)) * E' * y;       %Average of public variables
40 v = v + alpha_k*( y - E * Zcap);     %Update the dual variables
41 y1(2) = Zcap(1);                     %Update the public variable
       with PCM-B3
42 y1(3) = Zcap(3);                     %Update the public variable
       with PCM-A2
43
44 %Inner loop optimization
45 x = fmincon(@net1, block.Dwork(2).Data, Aineq, bineq, A1, beq1, lb1
       , ub1, nonlcon, options);
46 x1 = x;
47
48 %Local loss
49 loss1 = rB1*x1(1)^2 + rB2*x1(2)^2 + rL1*x1(3)^2 + rD1*i_D1^2;
50
51 %Store values for next iteration
52 block.Dwork(1).Data = loss1;
53 block.Dwork(2).Data = x1;
54 block.Dwork(3).Data = v';
55
56 end
```

Listing A.2: Augmented subsystem 1 loss

```
1
2 function a=net1(x)
3
```

```
4  global v;                                %Dual variables
5  global y1;                               %Public variables
6  alpha = 0.3;                             %Penalty vector
7
8  %The augmented lagrangian of the subproblem
9  a = rB1*x(1)^2 + rB2*x(2)^2 + rL1*x(3)^2 + rD1*(i_D1)^2  + v(1)*(x
       (2)-y1(2)) + v(2)*(x(3)-y1(3)) + (alpha/2)*(norm(x-y1))^2;
10
11 end
```